A Dynamic Clustering Mechanism with Load-Balancing for Flying Ad Hoc Networks

GODWIN ASAAMONING[1,2]**, (Student Member, IEEE), PAULO MENDES**[3,4]**, (Senior Member, IEEE), and NAERCIO MAGAIA**[1,5]

[1]COPELABS, Universidade Lusófona, 1749-024, Lisbon, Portugal ({godwin.anuork, naercio.magaia}@ulusofona.pt)
[2]Bolgatanga Technical University, UB-0964-8505 Sumbrungu, Ghana
[3]School of Communication, Architecture, Arts and Information Technologies, Universidade Lusófona, 1749-024 Lisbon, Portugal
[4]Airbus Central Research and Technology, Willy-Messerschmitt-Str 1, Taufkirchen, 82024 Munich, Germany (paulo.mendes@airbus.com)
[5]School of Engineering and Informatics, University of Sussex, Brighton BN1 9QJ, U.K.

Corresponding author: Godwin Asaamoning (godwin.anuork@ulusofona.pt).

ABSTRACT Flying Ad Hoc NETworks (FANETs) are expected to have a significant impact in several use-cases, from smart agriculture and cities, to mission critical scenarios. The recent surge in the use of FANETs is motivated by their adaptable and flexible behaviour in different scenarios (e.g. disaster-hit locations) allowing the usage of services that require information from remote locations, such as for assessment of damages, checking for survivors, or providing onsite views to assist rescue teams. While FANETs have been developed to provide such critical services, disseminating data with proper performance faces challenges due to inherent properties of FANETS, namely frequent wireless disconnections, intermittent available nodes, and dynamic topologies, mostly when facing an increasing number of deployed unmanned aerial vehicles. Aiming to tackle these challenges, we propose a new Dynamic Clustering Mechanism with Load-Balancing able to support efficient dissemination of data packets in FANETs while ensuring good reliability and scalability factors. The proposed solution is based on the combination of a new meta-heuristic optimization scheme, known as Political Optimizer, used to perform clustering while addressing limitations caused by topology changes, and a new Shannon entropy function implemented to address cluster fault tolerant and traffic overloads. Simulation results show that by combining our proposed model with standard position-based routing protocols, a higher number of end-to-end transmissions are ensured, while supporting an average packet delivery ratio of 97%, an average end-to-end delay of 0.225 seconds, and an average power consumption 37% lower than other state-of-the-art clustering protocols.

INDEX TERMS FANET, Dynamic Clustering, Load-Balancing, Routing.

I. INTRODUCTION

Flying Ad Hoc NETworks (FANETs) have emerged as a promising technology derived from Mobile Ad Hoc NETworks (MANETs) with a large set of applications [1]. However, it is well known that the number of Unmanned Aerial Vehicles (UAV) encompassed in a FANET, as well as the way they are organized, have a direct impact on the usefulness of the FANET to deliver data with high probability, low latency, and low power consumption over large areas.

Moreover, independently of the number of UAVs, the operation of FANETs come with significant challenges due to the dynamic topology, three-dimensional movement of UAVs, and intermittent wireless links [2]. Previous studies have shown that these challenges lead to network instability and high energy and communication overheads [3]. Recent proposals aimed to tackle some of these challenges in isolation, such as trying to reduce energy consumption by predicting the trajectory of UAVs, or implementing bio-inspired routing protocols to improve the efficiency of data transmission [4]. Yet, such approaches fail to address the overall challenge of ensuring efficient data transmission in large scale FANETs, while ensuring low levels of delay, packet loss and power consumption.

In such scenarios a cluster-based approach may allow the

creation of large scale FANETs while reducing power and communication overheads, if such approach is able to adapt to the mobility patterns of UAVs and to properly balance traffic among suitable Cluster Heads (CH). Our claim in this paper is that such a clustering based approach may ensure a good end-to-end performance when associated with standard routing protocols [5].

Some cluster-based solutions were proposed where CHs are selected to ensure an efficient management of clusters [6]. However, when the selection of CHs is done using Euclidean distance, research findings show poor performance mostly due to the noise associated with such measurements, thereby reducing the overall routing efficiency. In what concerns routing algorithms, several distribution-based optimization techniques have been deployed aiming to achieve good performance, e.g., routing based on movement awareness [7]. However, the increased computation needs of such approaches introduced additional overheads in latency and networking costs [8].

Keeping in mind the drawbacks of the existing methods, we argue in this paper that the most suitable method to route packets with high performance and low cost in FANETs is based on a novel clustering and load balancing mechanisms between selected cluster nodes. Following that, we propose a new Dynamic Clustering Mechanism with Load-Balancing able to support large scale fault-tolerant FANETs by: (i) dynamically grouping nodes into clusters taking into account their mobility patterns, and (ii) balancing traffic between CHs taking into account buffer occupancy. The aim being to support standard routing protocols, within and between clusters, to achieve a high packet delivery probability, low end-to-end delay, and low power consumption via a clustering mechanism that can support large sums of successful transmissions between a large number of UAVs.

Therefore, the contributions of this article can be described as follows:

- A new mobility aware dynamic clustering mechanism based on a socio-inspired meta-heuristic optimization algorithm aiming to support a stable and reliable operation of large scale FANETs, namely in what concerns data packet dissemination.
- An entropy-based optimal load-balancing mechanism to provide fault-tolerant data dissemination inside and between clusters.
- Through extensive simulations, we show that the proposed mechanisms can ensure a good end-to-end performance when simple position-based routing protocols are used within and between clusters.

The used research methodology aimed to investigate the two major contributions (clustering and load balancing) in an intertwined way, encompassing the analysis of requirements, designing of a proper combined solution, implementation of all needed components on a simulator (NS-3), and analysis of experimental results.

This article is organized as follows: Section II provides a brief analysis of prior approaches aiming to support an efficient operation of FANETs. In section III, we provide a description of the optimization algorithms that are used to devise the proposed solution. Section IV provides a description of the different components of the proposed Dynamic Clustering Mechanism with Load-Balancing, namely the mobility aware dynamic clustering, and the entropy-based load balancing. Section IV also provides a brief description of the standard routing protocols that are used in this study to ensure end-to-end connectivity within and between clusters. In section V we describe the experimental setup and conducts a performance evaluation of the proposed mechanism. Finally, section VI summarizes our findings and provide pointers towards future research directions.

II. RELATED WORK

Routing of data packets in FANETs is confronted with several challenges mainly due to the potential large network diameter, node dynamics, wireless intermittent links, and topology changes. As mentioned before, in this paper we argue that the best method to increase networking performance of FANETs, while reducing power cost, is by integrating suitable routing strategies with a clustering approach that is able to adjust considering node mobility and buffer occupancy on CHs.

Mindful of the significant envisioned role of UAV-FANET in future generation mobile networks and beyond, such as seamless and flexible provision of emergency communication coverage and information services [9], we find in the literature, barely few approaches that attempted to efficiently integrate clustering and optimization techniques to improve routing efficiency in FANETs. In one of such approaches, L. Ye et al. [3] propose a mechanism able to reduce end-to-end delay and increased throughput, but at the expense of high communication overheads. Most of similar solutions fail to achieve high performance with low costs due to the clustering mechanisms that are not aware of the network and traffic dynamics.

Some cluster-based routing solutions have been proposed aiming to address the dynamics found in FANETs. For example, Ali Khan et al. offered a bio-inspired clustering scheme aiming to provide efficient energy consumption and stable routing based on swarm optimization and a krill herd scheme to split the network nodes into several clusters [10]. In this proposal, the choice of CHs is done based on the residual energy levels of nodes. The dynamics of the network is considered by providing coherent alignment of cluster nodes based on the social behavior of insects, such as change and movement. Nevertheless, results show that the cluster lifespan is short-changed due to the constant topology changes, which affect the overall performance. You et al. [11] proposed a UAV clustering scheme that optimizes both coverage areas and mobile edge computing for FANET. The scheme provides coverage extension, prompt computation, and low transmission delay but failed to account on how the clusters and overall network reacts to the frequent topology changes and node mobility.

While trying to cope with network scaling, a mobility prediction clustering algorithm for managing network stability is proposed [12]. This algorithm explores the dictionary trie structure prediction scheme in sync with the link expiration time mobility model to manage the network cluster stability seamlessly. However, cluster formation is short lived causing more overheads mainly due to the frequent node dynamics and topology changes. S. Bhandari et al. [13] also proposed a mobility and location aware stable clustering scheme that estimate transmission efficiency by considering coverage probability and the size of clusters to cope with network scaling. In this case, the number of CHs is determined using a K-mean clustering formation scheme, and the maintenance of clusters is performed by considering the relative speed and distance of UAVs. But, this approach leads to buffer overflows and high overheads on energy and bandwidth spending, which may render the protocol unsuited for time-critical application scenarios that require sustained tasks' operations. To react to network topological changes, a self-organizing based clustering algorithm have been proposed for MANET aiming to aggregate node mobility based on zones in order to enhance network expansion and resilience [14]. The formation and maintenance of clusters is motivated by bird flocking behavior. Cluster heads are elected and oversees overall performances of cluster members. Communication between cluster members of different clusters is via a gateway node, which is responsible for facilitating communication between clusters. This study shows that the usage of self-organised methods aiming to reduce energy expenses in order to prolong network lifespan come with an increase of communication overheads, a drawback that we aim to tackle with the proposed Dynamic Clustering Mechanism with Load-Balancing.

Having as an objective to reduce energy spending, an energy-aware clustering mechanism was proposed to address limitations of high energy consumption and unreliable routing in FANETs [15]. In this approach, transmissions are adjusted to ensure optimal end-to-end packet transmissions resulting in reduced energy overheads. For this, a K-mean density clustering scheme is used to assist in selecting optimal CHs, thus increasing the lifespan of the CH while reducing operational overheads. However, the node mobility and link instability attributes of FANETs are not considered during the cluster formation process, hence adversely affecting the reliability of the proposed solution. A load-balancing and energy efficient clustering scheme which is based on swarm intelligence is introduced [16] and meant to provide communication needs during emergency communication. The scheme relies on metrics such as UAV residual energy, geo-location and distances of inter-cluster and intra-clusters to decide on the cluster head selection. Limitations of energy expending, traffic loads, coverage and network lifespan are tackled but the scheme accrues excess communication overheads.

In recent times, the utilization of meta-heuristic optimization inspired by nature, social and biological behaviors, have

variously been implemented for energy efficient clustering in wireless sensor networks (WSN). For instance, Mann et al. [17] proposed an improved meta-heuristic-based energy-efficient clustering scheme aiming to enhance the resolution exploration equation on its exploitation abilities together with an increase in its global convergence rates. The scheme thus retains a good optimal load balance between intra-cluster member as well as inter-cluster heads for sustained communication while utilizing less memory and energy overheads and high throughput. Inhere, cluster head is responsible for the management of cluster members as well as serving as a gateway to enable communication between a cluster member and a different cluster group member in the network. Similar meta-heuristic clustering approaches implemented [18] and [19] also aim at resolving load balancing concerns by creating effective inter-cluster routing to extend the network lifespan, improve throughput as well as minimize the traffic loads on gateways. Nonetheless, these techniques have mostly been utilized in WSN.

Based on the performed literature analysis, it is observed that existing clustering mechanisms partially provided solutions to the challenges faced by large FANETs, which are mainly caused by unreliable link connections, network complexity, scaling and instability. We, therefore, propose the investigation of a new dynamic clustering mechanism with load balancing to mitigate these constraints in large FANETs. The proposed clustering mechanism is able to work with any standard routing protocol.

Over the past decade, there has been several efforts towards proposing efficient routing solutions to mitigate the challenges faced by FANETs. For instance, opportunistic-based routing approaches [20] implement store-carry-and-forwarding schemes aiming to exploit nodes mobility to overcome the challenges imposed by intermittent wireless links in delay-tolerant networks as is the case of FANETS. However, since UAVs may follow a mobility pattern that may not be correlated with the social encounters assumed by opportunistic routing solutions, the efficiency of such routing protocols may be inconsistent [21]. Other approaches try to reduce the dependencies from mobility models by implementing a movement-aware routing approach [7], but with a surge on computation and latency overheads.

Several other approaches look at the routing challenge in FANETs as a distributed optimization problem by combining power control, delay and rate allocation [22]. However, such solutions do not consider the three-dimension factor of height and altitude, thus contributing to poor impacts on the overall efficiency of the schemes. Other proposals try to use fuzzy logic or learning processes to optimize routing in FANETs. Fuzzy logic schemes have been proposed to integrate the resource allocation and routing processes [23]. Such schemes show good fault tolerance, especially when number of nodes are increased, but the deployment of fuzzy logic for fitness computation increased the associated latency as there are many rules to be checked, which impacts on the routing performance. Some learning based multi-purpose optimization

solutions aim to select best neighbors by considering several factors such as geographic location, energy, mobility, and queuing delay [24]. However, these routing schemes do not consider the altitude and location of nodes, hence, impacting on the efficiency of the protocol.

Independently of the overall performance of routing optimization solutions, their complexity and the time needed to take decisions, may not be suitable for FANETs, which are constraints in terms of energy and communication opportunities. Hence, the hypothesis that is investigated in this paper is that an efficient dynamic clustering mechanism with load balancing allows the usage of standard routing protocols within and between clusters.

III. BACKGROUND

Our proposed mobility aware approach is based on a meta-heuristic optimization technique that has been very popular to solve global optimization problems. Generally speaking, meta-heuristic optimization procedures may be split into four groups: swarm-based, evolution-based, social-based, and physics-based algorithms [25].

Recently, Askari et al. [26] proposed a new meta-heuristic method called Political Optimizer (PO) that was inspired by the multi-staged process of politics and described a model to solve global optimization problems. The PO algorithm is the mathematical mapping of all the major stages of politics such as constituency allocation, party switching, election campaign, inter-party election, and parliamentary affairs.

It has been shown that the PO algorithm can solve classical engineering design problems, such as welded beam and speed reducer design [26]. Furthermore, the algorithm has an outstanding convergence speed performance, and a good exploration proficiency in early iterations, which makes it suitable for the coordination of clustered networks.

The experimental evaluations of the PO algorithm [26] demonstrated that it outperformed the particle swarm optimization (PSO) [27] and Grey Wolf Optimizer (GWO) [28] algorithms in what concerns agents exploitative, and exploration and convergence capabilities. The PO algorithm is known to stimulate agents search discovery and utilization by offering the agents with multiple options to update their search positions by splitting the population into parties and constituencies, where each member has the privilege to hold double roles. In this sense, search agents can improve their positions while interacting with the two best constituency and party candidates. Secondly, most algorithms utilize global best position to update the outcome of search agents whereas it is proven that the PO algorithm employs two best solutions to explore for any possible update of the position of agents. Further, a distinctive property of the PO algorithm is that the best positions of the individual search agents are always saved and referenced for future position updating, allowing the utilization of the most recent positions of each agent to uncover best search space regions. Additionally, the experimental results also prove that PO has strong convergence

to its global optima, as it retains its excellent routines when function shifting is applied in high dimensional functions.

TABLE 1. Comparison of position updating mechanisms and metric appraisals

Position updating mechanism	PO	PSO	GWO
Use of subgroup best	Yes	No	No
Use of global best position	No	Yes	Yes
Use of randomly selected solution	Yes	No	No
Preservation of solution records	Yes	Yes	No
Collaboration of better solutions for further improvement	Yes	No	No
Metric Appraisal	PO	PSO	GWO
Exploration ability	Good	Normal	Good
Exploitative ability	Good	Normal	Weak
Ability to avoid premature convergence	Normal	Weak	Average
Convergence speed	Fast	Slow	Slow
Capacity to converge on global best	Strong	Weak	Weak

In Table 1, we present a comparison of position updating mechanisms and metric appraisal, which illustrate the advantages of the PO algorithm over other meta-heuristic methods.

Therefore, our dynamic clustering mechanism is based on the PO algorithm. The PO process assumes that politicians always seek to optimize two different approaches: (i) each candidate aims to optimize its goodwill to win the election; (ii) each party aims to garner enough parliamentary seats. The algorithm consists of five stages: party formation and constituency allocation, election campaign, party switching, inter-party election, and parliamentary affairs. Such five phases are further divided in the following eight steps: (1) Population initialization, with the creation of different political parties and different constituencies; (2) Members of different political parties conduct activities within their constituency; (3) Party leaders and constituency winners are determined; (4) The positions of party members is updated according to the constituency winners; (5) The positions of party members is updated according to the party leaders; (6) The resultant positions are synthesized according to the position of the constituency winners and the party leaders; (7) Switching of parties between constituencies; (8) Election phase and reassign party leaders and constituency winners.

In a FANET scenario, the representatives of each constituency/cluster, i.e., cluster heads - CHs and substitute cluster head -SCHs, will divide the traffic load of their cluster. In the proposed solution, load balancing between CHs and SCHs is achieved by using a Shannon entropy algorithm.

The Shannon entropy is a widely known information theory concept used for providing insights into inferences of arbitrariness particularly in complex networks, physics, statistics, and dynamic systems [29]. Entropy is projected as a measure of diversity, disorder and uncertainty. Its probability distribution can be realized as a measure of uncertainty or randomness. For instance, to decide on differences or variations in the population of a given entity, entropy defines and incorporate diversity information, which allows for a study to be conducted to filter out the differences in the multi-paired

elements, on which foundation a trade off on collaboration can be reached.

IV. THE PROPOSED SOLUTION

This section details the two building blocks used to defined the Dynamic Clustering Mechanism with Load Balancing, namely the mobility aware dynamic clustering mechanism and the entropy-based load balancing mechanism. Towards the end of this section, we provide a brief description of the two standard routing algorithms that are considered in this study to drive the intra- and inter-cluster communication.

The overall goal is to keep the FANET network structured in a way that it allows an efficient routing of data packets between any pair of UAVs, or between UAVs and a base station in the ground, while improving performance parameters such as delivery probability, delay as well as power consumption. It is assumed that each UAV knows its position by means of a Global Positioning System (GPS), and is able of communicating by means of an omni-directional radio frequency system.

Figure 1, illustrates the architecture of our proposed solution, in which UAVs deployed in a certain geographical area are grouped into clusters. Each cluster has two representatives that are elected by means of a PO algorithm. Such representatives, CHs and SCHs share the traffic load in their cluster(s) by means of a Shannon entropy function, while serving as cluster gateways. The selected set of gateways (i.e., CHs) are able to communicate between them and with a set of base station(s) deployed in the ground and with the UAVs in their own clusters by means of an inter- and intra-cluster routing protocols, respectively.
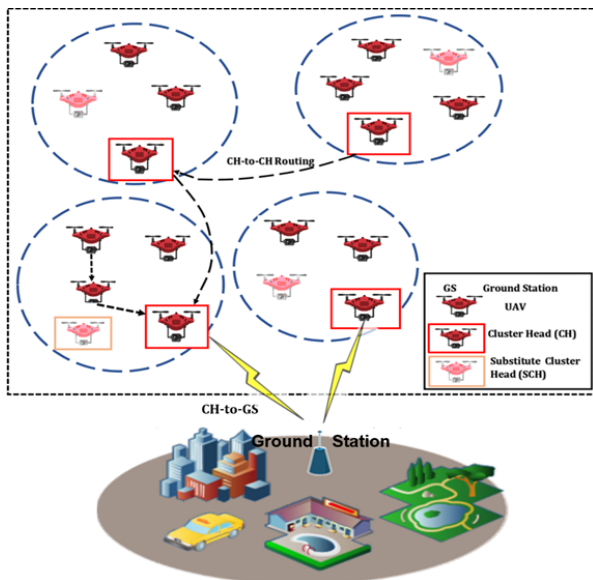

FIGURE 1. Architecture of proposed solution

A. MOBILITY AWARE DYNAMIC CLUSTERING

The proposed clustering mechanism aims to divide the network into separate groups of UAVs. In the current implementation, this algorithm is executed in a ground station based on information collected from each UAV, namely position (U_p), speed (U_s), moving direction (U_d), height variation (U_h), and link quality (U_L). Each deployed UAV is configured with the information about the geo-position of the ground station. Hence, throughout the lifespan of the network, the UAVs utilize a geo-position routing protocol to periodically update the ground station with information on their current position, speed, direction, height variation, and link quality. The usage of the proposed routing mechanism allows us to minimize used bandwidth, in comparison to using a simpler broadcast communication method. After providing this information, every individual UAV participates in the election process and a fitness value is computed for each UAV based on the number of votes that it gets by running the PO algorithm. The election process results in the election of a CH and an SCH per cluster is as shown in Figure 2, whilst in Table 2, we provide an index of the set of most used variables in this section.
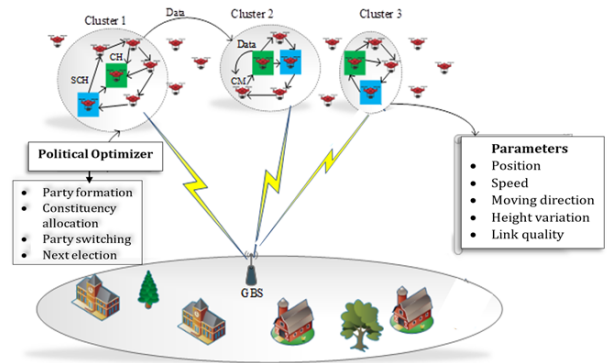

FIGURE 2. Mobility Aware Dynamic Clustering

TABLE 2. Index of the set of most used variables in this section

Variable	Meaning
ρ	Population (Entire members of all political parties)
ρ_i	Represents the ith party
p_i^j	Represents the jth member in the ith party
$\rho_{i,k}^j$	The kth dimension of the jth member in the ith party
c	Represents the group of all the constituencies
c_j	Represents the jth constituency
c_j^i	The ith party representative in the jth constituency
p_i^*	Denotes the leader of the ith party
c_j^*	Represents winner of the jth constituency
γ	Represents party switching rate
n	Represents the number of political parties, constituencies, and members in each party

The Mobility Aware Dynamic Clustering mechanism periodically starts by collecting updated information from each UAV, as previously stated. That information is used to associate the population of UAVs to n different parties of ρ based

on the similarities between them, which can be represented as Eq (1),

$$\rho = \{\rho_1, \rho_2, ..., \rho_n\} \tag{1}$$

Each party ρ_i comprise of n members (i.e., UAVs) as shown in Eq. (2),

$$\rho_i = \{p_i^1, p_i^2, ..., p_i^n\} \tag{2}$$

Each jth party member denoted as p_i^j is considered as a potential solution consisting of a d-dimensional vector, where d is the number of input variables of the problem to be solved as expressed in Eq. (3),

$$p_i^j = [p_{i,1}^j, p_{i,2}^j, ..., p_{i,d}^j]^T \tag{3}$$

After this, the position information of each UAV is used to collocate the different parties in a set of 3D dimensional spaces, which correspond to n constituencies in the PO process, as expressed in Eq (4),

$$c = \{c_1, c_2, ..., c_n\} \tag{4}$$

Every party member plays dual roles, being a member and a potential candidate for an election. Thus, if the jth member of each party in the jth constituency (c_j) participate in an election, then the winner is expressed as,

$$c_j = \{p_1^j, p_2^j, ..., p_n^j\} \tag{5}$$

After the election, the fittest member of a party is selected as the party leader, as is expressed in Eq. (6), where p_i^* designates the leader of the ith party and $f(p_i^j)$ calculates the fitness of p_i^j expressed as,

$$p_i^* = p_i^q \quad where \quad q = \underset{1 \leq j \leq n}{\operatorname{argmin}} f(p_i^j), \ \forall i \in \{1, ..., n\} \tag{6}$$

The collection of the group of all party leaders denoted by ρ^* is as presented in Eq. (7),

$$\rho^* = \{p_1^*, p_2^*, ..., p_n^*\} \tag{7}$$

The winners from all constituencies are known as parliamentarians. The group of all parliamentarians of the different constituencies denoted as c^* is expressed as,

$$c^* = \{c_1^*, c_2^*, ...c_n^*\} \tag{8}$$

At this stage, denoted as election campaign phase, the party and constituency leaders seek to influence their performance in the next election by using Eq. (9) and Eq. (10), which are mathematically modeled based on a position updating strategy to give insights on the previously held election from which candidates can harness to further improve the positions of party and constituency leaders. These equations are interchangeably used by assessing current and previous fitness values of a candidate. For instance, if the current fitness of the candidate is $f(p_i^j(t))$ represents an improvement and its past fitness was $f(p_i^j(t-1))$ represents a decline, first the position of the candidate is updated with reference to the party leader p_i^* and next with reference to the constituency winner c_j^*. Please note that Eq. (9) and Eq. (10) are used to compute the gain and lose of fitness, respectively. In these equations, r is a random variable number within the range of [0,1], while s^* holds two values, namely the value of the kth dimension of the party leader $p_{i,k}^*$ and that of $c_{j,k}^*$, which represents the constituency winner.

Therefore, to begin the process of updating positions on p_i^* and c_j^*, the current fitness of p_i^j is evaluated with respect to its previous fitness based on the following condition: $f(p_i^j(t)) \geq f(p_i^j(t-1))$. If last condition holds, Eq. (9) is initially used to update the position with respect to the party leader p_i^* of the ith party, in which case we substitute $p_{i,k}^j$ into s^*, inputs for k and r effected, and then substitute $p_{i,k}^j(t)$ into $p_{i,k}^j$ to compute the position update on party leader. After that, Eq. (9) is used to update position with respect to winner of the jth constituency c_j^* expressed based on Eq. (13) where the winner of the jth constituency is the representative of the ith party. Hence, $c_{j,k}^i$ is substituted into s^* and values of r and k respectively effected, and the substitution of $p_{i,k}^j$ into $p_{i,k}^j(t+1)$ accordingly made to compute position update on constituency winner. However, if the above mentioned condition of the evaluation with respect to the current and previous fitness values is false, Eq. (10) is used for the computation, following a similar set of considerations.

$$p_{i,k}^j(t+1) = \begin{cases} s^* + r(s^* - p_{i,k}^j(t)), & if \ p_{i,k}^j(t-1) \leq p_{i,k}^j(t) \leq s^* \ or \ p_{i,k}^j(t-1) \geq p_{i,k}^j(t) \geq s^*; \\ s^* + (2r-1)|s^* - p_{i,k}^j(t)|, & if \ p_{i,k}^j(t-1) \leq s^* \leq p_{i,j}^j(t) \ or \ p_{i,k}^j(t-1) \geq s^* \geq p_{i,k}^j(t); \\ s^* + (2r-1)|s^* - p_{i,j}^j(t-1)|, & if \ s^* \leq p_{i,k}^j(t-1) \leq p_{i,k}^j(t) \ or \ s^* \geq p_{i,k}^j(t-1) \geq p_{i,k}^j(t); \end{cases} \tag{9}$$

$$p_{i,k}^j(t+1) = \begin{cases} s^* + (2r-1)|s^* - p_{i,k}^j(t)|, & if \ p_{i,k}^j(t-1) \leq p_{i,k}^j(t) \leq s^* \ or \ p_{i,k}^j(t-1) \geq p_{i,k}^j(t) \geq s^*; \\ p_{i,k}^j(t-1) + r(p_{i,k}^j(t) - p_{i,k}^j(t-1)), & if \ p_{i,k}^j(t-1) \leq s^* \leq p_{i,k}^j(t) \ or \ p_{i,k}^j(t-1) \geq s^* \geq p_{i,k}^j(t); \\ s^* + (2r-1)|s^* - p_{i,j}^j(t-1)|, & if \ s^* \leq p_{i,k}^j(t-1) \leq p_{i,k}^j(t) \ or \ s^* \geq p_{i,k}^j(t-1) \geq p_{i,k}^j(t); \end{cases} \tag{10}$$

To form clusters for the election of stable and reliable CH and SCHs a fitness evaluation value is computed on the metrics. For dynamic scenarios such as UAV-based FANET, the fitness value is calculated based on the characteristic input parameters of position, speed, moving direction, height variation and link quality as shown in Eq. (11),

$$f(p) = \{U_p, U_s, U_d, U_h, U_L\} \qquad (11)$$

The next step is party switching also known as balancing, exploring and exploitation. Here, an adaptive parameter γ is used, which is linearly decreasing from one to zero during the entire iterative process. This parameter is termed as the party switching rate, allowing every UAV selected with probability γ to exchange with the worst member of a randomly selected party, based on Eq. (12),

$$\text{where} \quad q = \underset{1 \leq j \leq n}{\arg\max} f(p_r^j) \qquad (12)$$

In the election phase, the fitness of all constituency candidates participating in an election is evaluated. For instance, the fittest of all candidates participating in the jth constituency is evaluated, and the fitness value of the winner is expressed as,

$$c_j^* = p_q^i, \qquad \text{where} \quad q = \underset{1 \leq i \leq n}{\arg\min} f(p_i^j) \qquad (13)$$

The party leader and constituency winner are respectively expressed based on Eq. (6) and Eq. (13).

Based on the winners of each of the constituencies, i.e., the representatives called members of parliament in the PO algorithm, clusters are formed and the first two winners in each constituency are selected as CH and SCH for each cluster. This election process is done based on several properties of each UAV, such as energy level (N_e), buffer state (N_b), distance towards a base station (N_d), distance towards representatives of other constituencies (N_c), and link quality (N_l), as expressed in Eq. (14). The distance from an UAV towards a base station can be obtained by using a probe mechanism, such as pinging the IP address of known base station(s),

$$CH = \{N_e, N_b, N_d, N_c, N_l\} \qquad (14)$$

After the election, the selected SCHs are used to balance traffic load in their clusters in cooperation with the local CHs, using an entropy-based mechanism described in the next section. At the end of the election process, the base station broadcasts the election result to all network members. Algorithm 1, presents the pseudo-code for the mobility-aware clustering mechanism.

B. ENTROPY-BASED STABLE LOAD BALANCING

Cluster members (CM) communicate with nodes outside the cluster through the CH. Each CH aggregates packets from member nodes and routes them to the CH of the cluster where the destination node is located. In situations in which

---

Algorithm 1: Mobility Aware dynamic clustering

Input: $(U_p, U_s, U_d, U_h, U_L)$, Population ($\rho$)
Output: Clusters
$\mathbf{I}nitialize \leftarrow$
$\quad U_p, U_s, U_d, U_h, U_L$, and ρ based on Eq. (1) to Eq. (5)
for *every UAV* **do**
$\quad$ Compute U_p, U_s, U_d, U_h, U_L, based on Eq. (11)
$\quad$ Compute fitness of each member p_i^j based on
$\quad\quad$ Eq.(6)
$\quad$ Calculate the set of party leaders ρ^* based on Eq. (7)
$\quad$ Calculate the set of constituency winners c^* based
$\quad\quad$ on Eq. (13) and (8)
$\quad t = 1;$
$\quad \rho(t-1) = \rho;$
$\quad f(\rho(t-1)) = f(\rho);$
$\quad \gamma = \gamma_{max};$
$\quad$ **while** $t \leq T_{max}$, **do**
$\quad\quad \rho_{temp} = \rho;$
$\quad\quad f(\rho_{temp}) = f(\rho);$
$\quad\quad$ **for** *each* $\rho_i \in \rho$ **do**
$\quad\quad\quad$ **for** *each* $p_i^j \in \rho_i$, **do**
$\quad\quad\quad\quad p_i^j =$ElectionCampaign$(p_i^j, p_i^j(t-$
$\quad\quad\quad\quad 1), p_i^*, c_j^*);$
$\quad\quad$ Party Switching $(\rho, \gamma);$
$\quad\quad$ Evaluate worse fitness values for each member
$\quad\quad\quad$ based on Eq. (12)
$\quad\quad$ Calculate the set of constituency representatives c^*
$\quad\quad\quad$ based on Eq. (13) and (8)
$\quad\quad$ Parliamentary Affairs $(c^*, \rho);$
$\quad\quad \rho(t-1) = \rho_{temp};$
$\quad\quad f(\rho(t-1)) = f(\rho_{temp});$
$\quad\quad \gamma = \gamma - \gamma_{max}/T_{max};$
$\quad\quad t = t + 1;$
//Cluster head selection
Initialize clusters $\leftarrow N_e, N_b, N_d, N_c, N_l$
for *each cluster member,* **do**
$\quad$ Compute N_e, N_b, N_d, N_c and $\quad N_l$,
$\quad$ based on Eq. (14)
$\quad$ Select CH
$\quad$ Select SCH // substitute cluster head

---

the cluster has a large number of nodes or when the existing nodes generate a large amount of traffic, there is the possibility of overloading the buffer of the CH, resulting in increased latency and unwanted re-transmission of packets. In order to avoid such situation, a load balancing mechanism is implemented to split traffic between the CH and the SCH, as illustrated in Figure 3. The proposed load balancing mechanism is based on the Shannon entropy function.

Shannon entropy gives the concept of information loss that is convex-linear and continuous. The function of Shannon entropy is expressed in Eq. (15), where S represents a measure of choice, c represents a constant random variable that is a
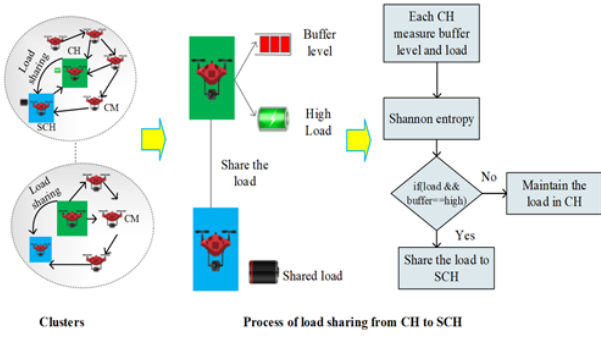
FIGURE 3. Entropy based Load Balancing

measure of uncertainty associated with a random variable, and L represents the current load of the node. The equation constitutes the discrete random variables $e \epsilon E$ considered by the probability $P(E)$.

$$S(l) = -c\Sigma_{l\epsilon L}P_{i-Th}(l)logP_{i-Th}(l) \qquad (15)$$

Eq. (16) represents the improved form of the Shannon entropy to a random variable, where B represents the buffer state of the current CH. The Shannon entropy function generates the threshold value based on the current load and buffer state of the CH. Suppose the load and buffer exceed the threshold value. In this case, the extra load is shared with the SCH. This decision is broadcast in the cluster, allowing sources to diverted traffic and start sending packets to SCH, resulting in stable load balancing and reducing the overhead caused due to the buffer overload.

$$S(l,b) = -c\sum_{l\epsilon L}\sum_{b\epsilon B}P_{i-Th}(l,b)logP_{i-Th}(l,b) \qquad (16)$$

The overall pseudo-code process to balance traffic load of a cluster between the CH and the SCH is as shown in Algorithm 2.

Algorithm 2: Load Balancing

Input: No. of cluster heads
$\qquad (CH_1, CH_2, ...CH_n), L, B$
Output: Balancing load
Initialize $\leftarrow (CH_1, CH_2, \ldots CH_n), L, B$
for *each node* **do**
$\quad$ Get L, B
$\quad$ Compute threshold based on Eq. (16)
$\quad$ **if** *(L&&B > Threshold)* **then**
$\quad\quad \llcorner$ Load is shared with SCH

C. ROUTING
As previously stated, this work investigates the hypothesis that the proposed Dynamic Clustering Mechanism with Load Balancing can be used with standard routing protocols within and between clusters. Here, we provide a brief description on

how to route packets within and between clusters following a position-based routing approach.

The basic assumption is that UAVs know their own geographical position and periodically exchange information with their neighbours. Elected CHs and SCHs identify themselves as such in these periodic broadcasts. In our study, it is also assumed that nodes have updated information about the geographic position of destination nodes. Please note that this is a basic assumption of position based routing protocols. Nonetheless, being out of scope of this paper, several proposals define a system able of keeping geo-position information up to date [30].

Here, intra-cluster routing is facilitated by a simple rank-based algorithm used to allow nodes to broadcast data packets towards a destination node. Such data packets include the position of the destination node, CH or SCH. After receiving data packets, nodes calculate their ranking based on the quality of their wireless link L_q, the length of their packet queue Q_l, their speed N_s, location N_l, direction N_d, the minimum of their distance towards the CH/SCH D_c, and the destination D_d. The computed rank is used to configure a random time interval after which the node transmit the packet, if no other transmission is overheard. This process helps to reduce bandwidth usage. Higher reliability is easily achieved by allowing a certain number of nodes to transmit packets even after overhearing a configured number of transmissions.

If the destination is in another cluster (i.e., the destination is faraway than the closest CH), packets are sent to the CH or SCH. At this point a position-based routing protocol is used to forward packets between CHs, until they reach the CH that is closest from the destination node. In this paper, we consider a Geographic Perimeter Stateless Routing (GPSR) based protocol, since it is widely adopted for vehicular networks [31]. In the operation of this protocol, every node maintains the knowledge of its one hop neighbors. Each node (i.e., CHs and SCHs) participating in the routing process selects as next-hop hop, the CH or SCH that is nearest to the destination. The locations of neighbors are obtained by exchanging periodic messages among nodes, as mentioned before.

When packets reach the CH or SCH that is closest to the destination, as a results of the inter-cluster routing process, they are routed inside the destination cluster towards the position of the destination based on the ranking algorithm.

V. PERFORMANCE EVALUATION
In this section, an evaluation of the proposed Dynamic Clustering Mechanism with Load-Balancing (DCM with Load-Balancing) is performed, aiming to demonstrate its effectiveness in terms of increasing the percentage of covered UAVs as well as the number of end-to-end transmissions over a set of created clusters. While the former aims to express the efficiency of the proposed PO based clustering method, the latter aims to show the efficiency of balancing load between the CH and SCH based on the entropy based mechanism. The performance of the proposed mechanism is compared

with the Stable Clustering Scheme (SCS) [13]. The SCS algorithm is a location-based K-means clustering mechanism integrated with the mobility and relative locations of UAVs to enhance performance. First, the location and mobility of UAVs are used to optimize deployment for maximum coverage performance by ensuring no UAV fields are overlapped with another while making sure that each UAV is attached to an identified CH. Cluster creation and maintenance is performed using K-means clustering approach followed by election of CHs. The overall goal is to strengthen the network performance and reliability.

Even if a clustering mechanism is able to cover a large percentage of deployed UAVs, the characteristic of the created clusters (e.g., diameter, density, and the average distance between UAVs) has a significant impact on data transmission in the FANET. Hence, besides evaluating the performance of the proposed DCM with Load-Balancing mechanism, our aim is also to assess the usefulness of the created clusters, i.e., the elected CHs and SCHs, in supporting a successful routing in terms of packet delivery probability, end-to-end delay and power consumption.

To analyze how efficient clusters are created and maintained by the proposed DCM with Load-Balancing mechanism in support of the process of routing packets end-to-end, we compare the performance of an intra- and inter-cluster position-based routing scheme (c.f. Section IV-C) with state-of-the-art routing solutions, namely the Energy-Efficient Opportunistic Routing Protocol (EEOR) [32] and the Smart IoT Control-Based Nature Inspired Energy Efficient Routing Protocol NIEEOR [33]. Whereas, the EEOR algorithm considers the prediction of UAV location based on node trajectory metrics before performing routing aiming to reduce packets' re-transmissions and energy spending. The NIEEOR algorithm introduced an energy stabilizing limit with the goal to select only nodes with higher energy levels than the identified threshold as relay nodes aiming to improve network quality of service.

The performance evaluation is done based on simulations carried out in the NS-3 simulator to analyse various performance metrics. Communications were performed using the 2.4 GHz frequency range of the IEEE 802.11n standard [34] under the Rayleigh channel model, used to calculate the channel fading characteristics. The simulation area is set to $1000m \times 1000m \times 880m$ with one ground base station for bi-direction communication, while UAVs are placed in 3D spaces. Their mobility is based on random way point (RWP) movement model. To get the most realistic setting for nodes progressive speed variations and direction, the JAVA code for RWP model was imported to enable the setting of moving parameters on the UAVs [32]. The simulation area was chosen to reflect the performed scenario of an increasing number of UAVs between 10 and 100. Our assumption is that the benefits of clustering are not so evident for smaller networks. The packet size is set to 600 bytes with an overall simulation time of 100 seconds. We provide a summary of the system configuration, as well as the parameters used in

the simulations in Table 3 and Table 4, respectively.

TABLE 3. System Configuration

	Processor	Pentium Dual Core and Above
Hardware Specification	RAM	8GB
	Hard Disk	60GB
Software Specification	Simulation Tool	NS-3 (version 3.26)
	OS	32-bit Ubuntu 14.4 LTS

TABLE 4. Simulation parameters

Parameters	Description
Simulation area	$1000m \times 1000m \times 880m$
Number of UAVs	10-100
Number of ground base station	1
Data rate	36 Mbps
Size of a message packet	600 bytes
Time to live	3h
Mobility	Random waypoint
Speed	30-50 m/s
Channel model	Rayleigh model
Transmission radius	1000m
Simulation time	100s
Transmission latency	20ms
Bandwidth	1MHz
GPS file	Route1.gps, route3.gps
Maximum energy	20250 J
Message interval	20ms-30ms

A. PERCENTAGE OF COVERED UAVs

The percentage of covered UAVs is an important metric for measuring the efficiency of a clustering process, since it is influenced by the parameters considered for the formation of clusters. In this context Figure 4 shows the results of a comparison of DCM with Load-Balancing and SCS with respect to the number of clusters that are created when facing an increasing number of UAVs. As showed in Figure 4 the proposed DCM with Load-Balancing mechanism covers a higher number of UAVs than SCS, being able to cover all the deployed UAVs with twenty clusters, while with SCS the same number of clusters only cover 80% of all UAVs. The better performance of the DCM with Load-Balancing scheme is due to its mobility awareness based on the factors such as position, speed, moving direction, height variation, and link quality.

Table 5 presents the numerical comparison between the the DCM with Load-Balancing and the SCS clustering mechanism in what regards the percentage of UAVs that are covered by the formed clusters. The DCM with Load-Balancing approach covers an average of 70% of UAVs, whereas the SCS
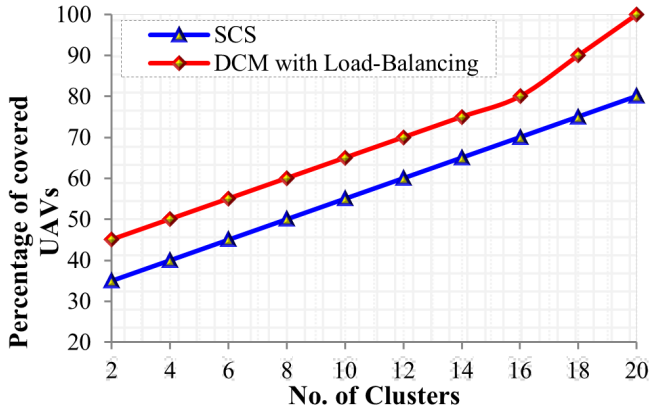
FIGURE 4. Number of clusters vs. Percentage of covered UAVs

approach covers only 57% of UAVs. The main reason for this differential is the fact that the SCS approach implements a K-means clustering approach in which the number of clusters needs to be determined a priori. This makes SCS unable to react swiftly to the dynamic topology of UAVs resulting in inefficient clustering.

TABLE 5. Analysis of Percentage of covered UAVs

Technique	Average percentage of covered UAVs
SCS	57.5±3
DCM with Load-Balancing	70.2±1

B. NUMBER OF END-TO-END TRANSMISSIONS

The number of supported end-to-end transmissions is an essential metric used to determine the efficiency of a clustering process.
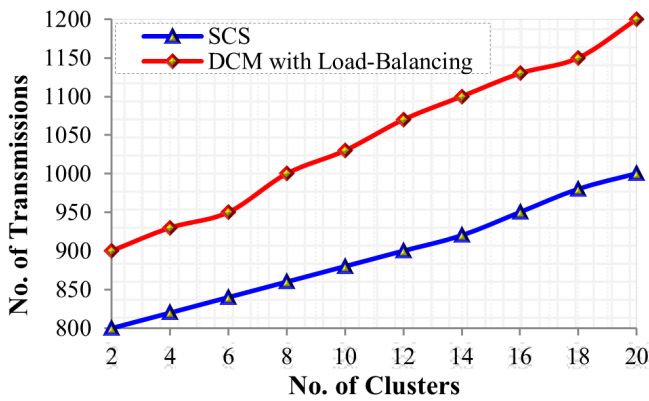

FIGURE 5. Number of clusters vs. Number of transmissions

A higher number of end-to-end transmissions reflects the efficiency of the clustering process in supporting the forwarding of data packets from source to destination, in what concerns potential bottlenecks between clusters. As illustrated in Figure 5 a higher number of end-to-end transmissions are

achieved when clustering the network based on the DCM with Load-Balancing approach, when compared to SCS. This higher number of transmissions is due not only to the PO clustering process, but mostly to the election of CHs and the usage of the entropy-based scheme to balance the load between the elected CH and SCH. The proposed entropy-based load balancing mechanism helps to ease the limitations brought by network instability and by the potential congestion of CHs. On the other hand, the SCS approach performed K-means clustering, which does not address the dynamic topology of flying networks, resulting in congestion, which further reduces the number of transmissions.

TABLE 6. Analysis of Number of transmissions

Technique	Number of transmissions
SCS	895.2±3
DCM with Load-Balancing	1046.8±1

Table 6 provides the numerical comparison of the number of transmissions achieved when the DCM with Load-Balancing and SCS approaches are used. The former is able of supporting an increased number of transmissions up to an average of 1046 transmissions, whereas the SCS approach is able to sustain only an average of 895 transmissions, mostly due to its incapability of balancing loads between clusters. The higher number of successful end-to-end transmissions supported by the DCM with Load-Balancing approach contributes to the efficiency of the routing process as shown in the analysis of packet delivery ratio, end-to-end delay and power expending performed in the following subsections.

C. PACKET DELIVERY RATIO

The packet delivery ratio is the ratio between the total number of packets received by the destinations and the total number of packets transmitted by the sources. A higher packet delivery ratio denotes a good routing performance.
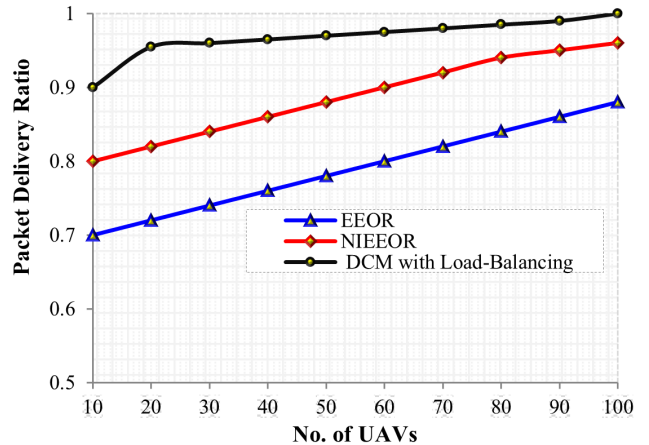

FIGURE 6. Number of UAVs vs. Packet delivery ratio

In this sense, Figure 6 illustrates the efficiency of the position-based routing strategy supported by the DCM with

Load-Balancing mechanism, in comparison with the EEOR and NIEEOR routing protocols. Figure 6 shows that while the EEOR and NIEEOR routing protocols have a similar performance when facing an increasing number of UAVs (with a differential of circa 10%), routing packets based on the DCM with Load-Balancing mechanism is able of achieving a higher percentage of delivered packets, even with a small number of deployed UAVs: from 90% with only 10 UAVs to close to 100% with 70 UAVs. These results show that an efficient clustering and local balancing procedure can support a good performance of simple routing protocols, such as the ranking intra-cluster and the geographical inter-cluster routing used in this study. On the other hand, performing routing over the complete network, by using the EEOR and NIEEOR routing protocols, results in a reduced packet delivery ratio, mostly due to the inefficient convergence, which implies an scalability issue, that is shown to be mitigated by clustering the network.

TABLE 7. Analysis of Packet delivery ratio

Technique	Average packet delivery ratio
ERROR	0.79± 0.05
NIEEOR	0.887± 0.03
DCM with Load-Balancing	0.968± 0.02

Table 7 provides the numerical comparison of the average packet delivery ratio when the routing process is based on the proposed DCM with Load-Balancing mechanism, as well as when the EEOR and NIEEOR routing protocols are used to route packets, in respect to number of UAVs. Table 7 shows that with a network varying from 10 to 100 UAVs, routing based on the proposed DCM with Load-Balancing mechanism achieves an average packet delivery ratio of 97%, which is 9% and 18% higher than using the NIEEOR and the EEOR routing protocols, respectively.

D. END-TO-END DELAY
The end-to-end delay corresponds to the total time needed to transmit a packet between a source and a destination. Figure 7 illustrates the average end-to-end delay achieved when routing packets based on our proposed mechanism, as well as the NIEEOR and the EEOR routing protocols. As shown, the average end-to-end delay increases linearly with the number of UAVs when the NIEEOR and the EEOR routing protocols are used. This is mostly due to the fact that such approaches perform routing by considering only the energy and speed without considering hop count, resulting in increased end-to-end delay.

When using the proposed DCM with Load-Balancing mechanism, the average end-to-end delay increases with the number of deployed UAVs, but with a smaller factor than when the NIEEOR and the EEOR routing protocols are used. On the one hand, this is due to the execution of a mobility-aware clustering and the selection of CHs based on link quality. On the other hand, the good performance of routing based on the proposed DCM with Load-Balancing
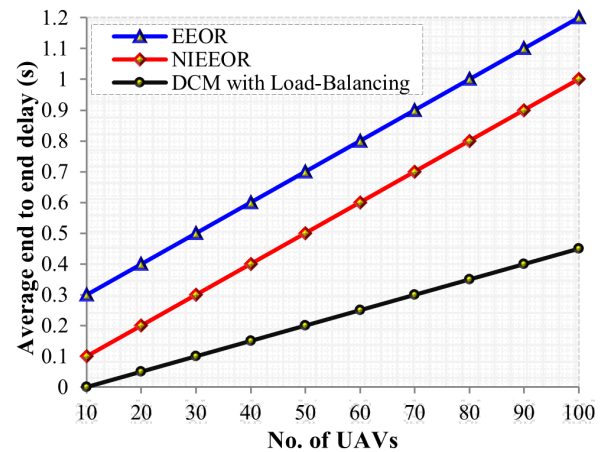

FIGURE 7. Number of UAVs vs. Average end-to-end delay

mechanism is due to the usage of an inter-cluster routing protocol only between a reduced number of UAVs (CHs and SCHs), thereby contributing to the reduced end-to-end delay.

TABLE 8. Analysis of Average end-to-end delay

Technique	Average end-to-end delay
ERROR	0.75± 0.3
NIEEOR	0.55± 0.2
DCM with Load-Balancing	0.225± 0.1

The numerical comparison of the end-to-end delay obtained with a routing protocol incorporated with the proposed mechanism, as well as the NIEEOR and EEOR routing protocols is presented in Table 8. The average end-to-end delay of our approach is as low as 0.225 seconds, whereas the NIEEOR and EEOR protocols may reach average end-to-end delays exceeding 0.55 seconds.

E. POWER CONSUMPTION
The power consumption of UAVs is an important metric to be considered to validate the energy efficiency of a clustering approach. Therefore, an efficient clustering approach should be able of creating clusters that allow UAVs to communicate without spending a lot of energy, for instance, by grouping UAVs that are at a close distance from the CH. In this context, Figure 8 shows the power consumption encompassed in data transmission activities with an increasing number of UAVs. Here, the position-based routing scheme incorporated with the proposed DCM with Load-Balancing mechanism is also compared with the EEOR and NIEEOR routing protocols.

The power consumption involved in data transmissions in a network clustered with the DCM with Load-Balancing mechanism is much lower than when using the the EEOR and NIEEOR routing protocols. The major reason for the higher power consumption of the EEOR and NIEEOR protocols is related to the higher usage of re-transmissions due to a low packet delivery ratio and a high packet drop count. On the other hand, the DCM with Load-Balancing mechanism
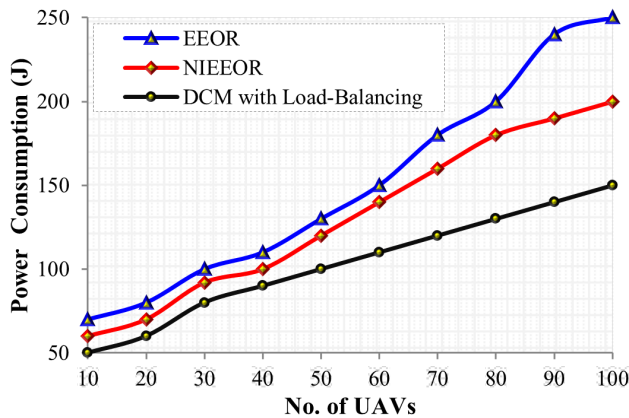
FIGURE 8. Number of UAVs vs. Power consumption

supports routing with a lower number of re-transmissions due to its awareness of UAVs mobility. Moreover, the increased power consumption of EEOR and NIEEOR protocols reduces the network lifetime, hence its usefulness.

TABLE 9. Analysis of Power consumption

Technique	Average power consumption in joules
ERROR	151±5
NIEEOR	131.2±3
DCM with Load-Balancing	103± 2

Table 9 presents the numerical comparison of the average power consumption of our approach, and the EEOR and NIEEOR routing protocols. The average power consumption when the DCM with Load-Balancing mechanism is used is of about 103 J. In contrast, the EEOR and NIEEOR routing protocols consume an average of 151J and 131J, respectively. These findings show that our approach can support data transmission in FANETs with a higher energy-efficiency than just applying existing routing protocols over the complete network.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we propose a new Dynamic Clustering Mechanism with Load-Balancing aiming to tackle the challenges faced by FANETs, namely intermittent available wireless links and dynamic topologies. The proposed solution guarantees efficient dissemination of data packets in FANETs by dynamically grouping UAVs into clusters by means of a Political Optimizer algorithm that takes into account the mobility patterns of UAVs, and by means of balancing traffic between cluster heads and secondary cluster heads based on a Shannon entropy function that takes into account transmission queues and network load. The proposed solution is capable of providing sufficient support for efficient dissemination of data packets in large FANETs while ensuring exceptional wireless reliability and scalability factors. Simulation results show that the proposed Dynamic Clustering Mechanism with Load-Balancing ensures an efficient clustering process by

covering in average 70% of UAVs. Also, experimental results show that by combining the proposed mechanism with standard position based routing protocols, we can achieve better performances than prior art in terms of energy consumption, guaranteed high packet delivery ratios and reduced network transmission delay.

As future work, we aim to perform the validation of the proposed clustering mechanism on a real-world testbed.

REFERENCES

[1] R. A. Nazib and S. Moh, "Routing protocols for unmanned aerial vehicle-aided vehicular ad hoc networks: A survey," *IEEE Access*, vol. 8, pp. 77 535–77 560, 2020.

[2] F. Wang, Z. Chen, J. Zhang, C. Zhou, and W. Yue, "Greedy forwarding and limited flooding based routing protocol for uav flying ad-hoc networks," in *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE, 2019, pp. 1–4.

[3] L. Ye, Y. Zhang, Y. Li, and S. Han, "A dynamic cluster head selecting algorithm for uav ad hoc networks," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2020, pp. 225–228.

[4] A. Salam, Q. Javaid, and M. Ahmad, "Bioinspired mobility-aware clustering optimization in flying ad hoc sensor network for internet of things: Bimac-fasnet," *Complexity*, vol. 2020, 2020.

[5] G. Asaamoning, P. Mendes, D. Rosário, and E. Cerqueira, "Drone swarms as networked control systems by integration of networking and computing," *Sensors*, vol. 21, no. 8, p. 2642, 2021.

[6] A. Raza, M. F. Khan, M. Maqsood, B. Haider, and F. Aadil, "Adaptive k-means clustering for flying ad-hoc networks." *KSII Transactions on Internet & Information Systems*, vol. 14, no. 6, 2020.

[7] K. Kumar, S. Kumar, O. Kaiwartya, P. K. Kashyap, J. Lloret, and H. Song, "Drone assisted flying ad-hoc networks: mobility and service oriented modeling using neuro-fuzzy," *Ad Hoc Networks*, vol. 106, p. 102242, 2020.

[8] S. Rezwan and W. Choi, "A survey on applications of reinforcement learning in flying ad-hoc networks," *Electronics*, vol. 10, no. 4, p. 449, 2021.

[9] J. Zhou, D. Tian, Z. Sheng, X. Duan, and X. Shen, "Joint mobility, communication and computation optimization for uavs in air-ground cooperative networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 3, pp. 2493–2507, 2021.

[10] A. Khan, F. Aftab, and Z. Zhang, "Bicsf: Bio-inspired clustering scheme for fanets," *IEEE Access*, vol. 7, pp. 31 446–31 456, 2019.

[11] W. You, C. Dong, X. Cheng, X. Zhu, Q. Wu, and G. Chen, "Joint optimization of area coverage and mobile-edge computing with clustering for fanets," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 695–707, 2020.

[12] C. Zang and S. Zang, "Mobility prediction clustering algorithm for uav networking," in *2011 IEEE GLOBECOM Workshops (GC Wkshps)*. IEEE, 2011, pp. 1158–1161.

[13] S. Bhandari, X. Wang, and R. Lee, "Mobility and location-aware stable clustering scheme for uav networks," *IEEE Access*, vol. 8, pp. 106 364–106 372, 2020.

[14] F. Aftab, Z. Zhang, and A. Ahmad, "Self-organization based clustering in manets using zone based group mobility," *IEEE Access*, vol. 5, pp. 27 464–27 476, 2017.

[15] F. Aadil, A. Raza, M. F. Khan, M. Maqsood, I. Mehmood, and S. Rho, "Energy aware cluster-based routing in flying ad-hoc networks," *Sensors*, vol. 18, no. 5, p. 1413, 2018.

[16] M. Y. Arafat and S. Moh, "Localization and clustering based on swarm intelligence in uav networks for emergency communications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8958–8976, 2019.

[17] P. S. Mann and S. Singh, "Improved metaheuristic based energy-efficient clustering protocol for wireless sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 142–152, 2017.

[18] Y. Han, G. Li, R. Xu, J. Su, J. Li, and G. Wen, "Clustering the wireless sensor networks: a meta-heuristic approach," *IEEE Access*, vol. 8, pp. 214 551–214 564, 2020.

[19] Z. Wang, H. Ding, B. Li, L. Bao, and Z. Yang, "An energy efficient routing protocol based on improved artificial bee colony algorithm for wireless sensor networks," *IEEE Access*, vol. 8, pp. 133 577–133 596, 2020.

[20] W. Moreira and P. Mendes, "Survey on opportunistic routing for delay-disruption tolerant networks," *SITILABS Technical Report SITI-TR-11-02*, 2010.

[21] W. Moreira, P. Mendes, and S. Sargento, "Assessment model for opportunistic routings," in *in Proc. of IEEE Latincom*, oct 2011.

[22] S. Wen, L. Deng, and Y. Liu, "Distributed optimization via primal and dual decompositions for delay-constrained fanets," *Ad Hoc Networks*, vol. 109, p. 102288, 2020.

[23] A. Khanna, J. J. Rodrigues, N. Gupta, A. Swaroop, and D. Gupta, "Local mutual exclusion algorithm using fuzzy logic for flying ad hoc networks," *Computer Communications*, vol. 156, pp. 101–111, 2020.

[24] J. Liu, Q. Wang, C. He, K. Jaffrès-Runser, Y. Xu, Z. Li, and Y. Xu, "Qmr: Q-learning based multi-objective optimization routing protocol for flying ad hoc networks," *Computer Communications*, vol. 150, pp. 304–316, 2020.

[25] F. Fausto, A. Reyna-Orta, E. Cuevas, A. G. Andrade, and M. C. Perez, "From ants to whales: metaheuristics for all tastes," *Artificial Intelligence Review*, Jan 2019.

[26] Q. Askari, I. Younas, and M. Saeed, "Political optimizer: A novel socio-inspired meta-heuristic for global optimization," *Knowledge-Based Systems*, vol. 195, p. 105709, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705120301350

[27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[28] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46–61, 2014.

[29] A. Lesne, "Shannon entropy: a rigorous notion at the crossroads between probability, information theory, dynamical systems and statistical physics," *Mathematical Structures in Computer Science*, vol. 24, no. 3, 2014.

[30] N. Magaia, G. Mastorakis, C. Mavromoustakis, E. Pallis, and E. K. Markakis, Eds., *Intelligent Technologies for Internet of Vehicles*, ser. Internet of Things. Cham: Springer International Publishing, 2021.

[31] B. Karp and H.-T. Kung, "Gpsr: Greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, 2000, pp. 243–254.

[32] Q. Sang, H. Wu, L. Xing, H. Ma, and P. Xie, "An energy-efficient opportunistic routing protocol based on trajectory prediction for fanets," *IEEE Access*, vol. 8, pp. 192 009–192 020, 2020.

[33] I. U. Khan, I. M. Qureshi, M. A. Aziz, T. A. Cheema, and S. B. H. Shah, "Smart iot control-based nature inspired energy efficient routing protocol for flying ad hoc network (fanet)," *IEEE Access*, vol. 8, pp. 56 371–56 378, 2020.

[34] T. Paul and T. Ogunfunmi, "Wireless lan comes of age: Understanding the ieee 802.11 n amendment," *IEEE Circuits and systems magazine*, vol. 8, no. 1, pp. 28–54, 2008.

PAULO MENDES is Expert in Network Architectures and Protocols at Airbus central research and technology in Munich, Germany. His research interests include self-organized wireless networks, information and service-centric networking, as well as in-network computing. Paulo is also an invited associate professor at University Lusofona, where he was associate professor from 2010 to 2019. Before joining Airbus, Paulo co-founded the COPELABS research centre (2010), the Senception start-up (2014), and the Internet architectures and networking research group at INESCTEC laboratory (2007). From 2003 to 2007 Paulo was senior researcher at NTT Docomo Euro-labs. In 2004, Paulo Mendes got his Ph.D. degree (summa cum laude) in informatics engineering from the University of Coimbra, while being a visiting scholar at Columbia University, New York, from 2000 to 2004. He is an IEEE senior member and an ACM member.

NAERCIO MAGAIA received the (5 year) B.Eng. degree in electrical engineering from Eduardo Mondlane University, Maputo, Mozambique, and the M.Sc. degree in communication networks engineering and the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico, University of Lisbon (ULisboa), Lisbon, Portugal.

His current research interests cover Computer Networking, the Internet of Thing, Network Security, and Artificial Intelligence.

● ● ●

GODWIN ASAAMONING is currently a junior researcher at the Cognitive and People-centric Computing Laboratory (COPELABS) Research and Development unit, and a PhD student at the New Media and Pervasive Systems programme of University Lusófona de Humanidades e Tecnologias in Lisbon, Portugal. His PhD work is in the area of Wireless Networking and communication for Autonomous Mobile Smart Camera Drones.

He received his M.Sc. degree in Information Technology from the Kwame Nkrumah University of Science and Technology, and a Bachelor of Education degree in Information Technology from University of Education, Winneba, both in Ghana. Before leaving for his PhD studies in 2018, Godwin was Head of Information and Communication Technology unit and Lecturer at Bolgatanga Technical University. He previously was a tutor at the Bagabaga College of Education in Tamale, and Sandema Senior High Technical School. At present, his research interests encompass: the Internet of Things, Wireless Communications, FANETs, and Distributed Systems. He is an IEEE student member.