Ad Hoc Networks 154 (2024) 103354

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

K-means online-learning routing protocol (K-MORP) for unmanned aerial vehicles (UAV) adhoc networks

Saifullah [a], Zhi Ren [a], Khalid Hussain [b], Muhammad Faheem [c,*]

[a] Chongqing Key Laboratory of Mobile Communications Technology School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400044, China
[b] Electro-Optical Communication, Department of Electrical Engineering, Technical University of Eindhoven (TU/e), Eindhoven, Netherlands
[c] School of Technology and Innovations, University of Vaasa, 65200 Vaasa, Finland

ARTICLE INFO

Keywords:
UAV networks
Routing protocols, Gauss markov 3D mobility model
k-means online learning

ABSTRACT

Unmanned Aerial Vehicles (UAVs) have become a hot topic due to their flexible architecture adopted in many wireless technologies. In UAV ad hoc networks, traditional routing protocols with a fixed topology are ineffective due to dynamic mobility and unstable paths. Therefore, the mobility patterns of UAVs challenge efficient and reliable routing in UAV networks. Traditional routing algorithms are often based on assumptions of static nodes and predetermined network topologies. Which are not suitable for the dynamic and unpredictable nature of UAV mobility patterns. To address this problem, this paper introduces a K-means online learning routing protocol (KMORP) scheme employing a Markov mobility model for UAV ad hoc networks. Initially, the proposed method utilizes a 3D Gauss Markov mobility model to accurately estimate UAV positions, while K-means online learning is adopted for dynamic clustering and load balancing. Designed for real-time data processing, KMORP is well suited for UAV ad hoc networks, quickly adapting to network environmental changes such as UAV mobility, interference, and signal degradation to ensure efficient data transmission and communication. This is achieved while reducing the overall communication overhead and increasing the packet delivery ratio(PDR%). In the routing phase, the proposed scheme employs inter-cluster forwarding nodes to transmit messages among different clusters. Extensive simulations demonstrate the performance of the proposed KMORP, showing a 38% better PDR compared to OLSR and over 50% less end-to-end(E2E) delay compared to typical K-Means. Furthermore, the proposed KMORP exhibited an average throughput of 955 kbps, showing a substantial improvement in network performance. The results underscore that the proposed KMORP outperforms existing techniques in terms of PDR, E2E delay, and throughput.

1. Introduction

Unmanned Aerial Vehicles (UAVs) have become a promising technology for various wireless communication applications. UAVs are flexible and can vary in size. They are ideal for different applications, such as military operations, goods delivery, traffic monitoring in urban areas, natural disaster and emergency rescue operations, and event live-streaming. UAV networks are infrastructure-less, self-organized, and decentralized networks that operate in three-dimensional (3D) spaces [1,2]. These networks utilize properties similar to those of Mobile Ad-hoc Networks (MANET) and Vehicular Ad-hoc Networks (VANET) to gather data from various locations and transmit that data to their destination via communication links. Because of the 3D-environment of UAV ad-hoc networks, it is crucial to count the UAVs' positions, movement, and direction [3,4]. Thus, different UAV routing protocols have been proposed to improve the performance of UAV ad hoc networks. Routing protocols directly affect the communication efficiency of multiple UAVs [5]. Traditional topology-based routing protocols sustain routing tables to perform the routing mechanism and forward data packets to the destination through the shortest path, usually counted hop by hop. Because hops are moving UAVs, efficient path optimization is essential [6,7]. However, UAVs are moving at a very fast speed. Hence, topology-based routing requires extra overhead to reinitiate route discovery, making it unsuitable for supporting the 3D nature of the UAV network [8,9]. To address this issue, position-based routing protocols utilize additional geographical information available through GPS

* Corresponding author.
E-mail address: muhammad.faheem@uwasa.fi (M. Faheem).

https://doi.org/10.1016/j.adhoc.2023.103354
Received 18 July 2023; Received in revised form 15 November 2023; Accepted 15 November 2023
Available online 18 November 2023

or other position-tracking services. A source UAV node forwards data packets to a destination UAV using position or geographic information [10,11].

Nonetheless, position-based routing schemes operate on local information instead of the entire network, resulting in significant delays when the source UAV node cannot find the proper coordination position of the destination node in one hop. Additionally, the high mobility of UAVs can cause nodes to lose connection or be out of the communication range, which may decrease packet delivery for the entire network [12]. Therefore, it is necessary to estimate the geographical locations of the entire network to determine the appropriate next hop toward the target. Hence, each UAV node sends data packets to the optimal neighboring UAV node present in its communication range towards the direction of the destination UAV [13,14]. The emergence of UAV-based networks introduces unique features and challenges compared with traditional MANET and VANET networks. Maintaining communication links between UAVs poses significant difficulties due to their fast movement along sparse trajectories. This creates new network requirements, such as 3D coordinates, stable connections, and high-speed mobility, which must be carefully considered [15]. As intermediate UAVs are responsible for transferring data from source to destination, a novel network environment is created, distinct from traditional MANETs/VANETs, which demand new mobility models, 3D coverage areas, and innovative strategies. The UAV network offers a 3D wireless network environment with free space for independent movement, presenting unique challenges for the research community, such as communication in a 3D environment due to high UAV mobility [16]. Routing the data from the source to the destination is challenging in an inter-UAV network [17]. Therefore, different mobility models were mathematically formulated and performed in simulation environments to provide realistic scenarios for complex networks.

Additionally, UAVs in wireless networks possess high mobility due to drones moving at different velocities in the air. In contrast, traditional network nodes travel on predefined paths, such as vehicles moving on roads and streets. Therefore, selecting an appropriate mobility model for a particular scenario in a UAV network is challenging [18].To enable effective communication between UAVs in a network, clustering-based UAV ad hoc routing schemes have emerged as promising solutions for enhancing the scalability and efficiency of UAV communication networks. These schemes divide the network into clusters, and a cluster-head UAV [19] controls each cluster. The cluster head manages communication within the cluster and relays messages to other clusters through inter-cluster communication. By utilizing clustering, these schemes reduce the overhead of routing protocols and improve the robustness and energy efficiency of the network. In this context, various clustering-based routing schemes have been proposed, each with advantages and limitations [20]. These schemes differ regarding the clustering algorithm used, the selection criteria for cluster heads, and the routing mechanism employed. Therefore, it is essential to carefully evaluate these schemes to identify the most suitable for a particular application. Overall, clustering-based UAV ad hoc routing schemes have shown great potential for enhancing the performance of UAV communication networks [21]. Their continued development and refinement are crucial for realizing the full potential of UAV technology. However, due to the high mobility of UAVs, these schemes suffer to form optimal clusters and face issues such as cluster overlapping or some UAVs are not part of any cluster in case they move away from the cluster range. Moreover, traditional cluster schemes cannot monitor the real-time locations of UAVs for clustering.

To address such issues, this paper introduces a k-means online learning routing protocol that monitors the real-time movement of UAVs in an ad-hoc network and forms clusters based on current data. Furthermore, this study adopted the Gauss Markov 3D mobility model to accurately count the position and velocity of each UAV in a 3D environment. The Gauss-Markov 3D Mobility Model brings multiple benefits to 3D routing in UAV networks, such as simulating realistic 3D movement patterns, incorporation of correlation and autocorrelation in UAV movement, customizable parameters, and efficient computation. In addition, K-means online learning limits messaging spreading, reduces the network load, and enhances the efficiency of the entire network. This paper outlines the main contributions below.

1. The proposed approach improves the performance and reliability of UAV ad hoc routing protocols by leveraging the strengths of the 3D Gauss Markov model and K-means online learning. The approach enables UAVs to select the optimal route for data transmission based on real-time data to estimate the UAV positions and communication channel parameters, resulting in efficient communication between the UAVs and efficient utilization of network resources.
2. The proposed scheme reduces the load among UAVs in the network, ensuring that network resources are efficiently utilized by dividing the network into different clusters. This can help avoid congestion and improve the overall performance of the network.
3. The K-means online learning routing protocol processes data in real time, making it efficient for UAV ad hoc networks where UAVs are constantly moving and the network topology is changing. Furthermore, this approach can minimize the communication between nodes with the help of clustering, thereby reducing the overall network load.
4. Improved accuracy: K-means online learning can improve the accuracy of clustering results over time as more data is processed. The algorithm can adapt to changes in the data distribution, resulting in more accurate clustering results.

This article is structured as follows: Section 1 provides an introduction to the paper and outlines its contributions. Section 2 presents a comprehensive review of the existing literature on routing protocols for UAV networks. Section 3 presents the mathematical formulation of the proposed scheme, a description of the algorithm, and an explanation of its working principle. Section 4 provides an analyzes of results from the proposed scheme, utilizing performance evaluation parameters and simulation data. Finally, Section 5 concludes the article with a discussion of future work and provides a list of references.

2. Related work

In this section, we have investigated the literature review of previous research work and highlighted the technical schemes mentioned below. The article [22] discussed the challenges and solutions for designing a routing protocol for unmanned aerial vehicles (UAVs) in a flying ad-hoc network (FANET), with applications ranging from disaster management to smart agriculture. Moreover, the authors proposed a hybrid bio-inspired model for destination-aware routing in UAV networks, which uses three optimization techniques (PSO, GA, and GWO) to minimize the distance of the routing path, maximize energy efficiency levels, and ensure fault tolerance. In [23], the authors proposed an integrated host- and content-centric routing mechanism for Unmanned Aerial Vehicle (UAV) swarms. The mechanism harnesses the advantages of both routing mechanisms and reuses the routing information of stable paths in a host-centric manner to reduce flooding for path exploration. Route failure detection and re-routing are content-centric to adjust to topology dynamics. The authors [24] focused on optimizing the performance of unmanned aerial vehicle (UAV) swarms using the particle swarm optimization algorithm. Initially, the authors proposed a solution using the Cauchy particle swarm optimization method, which they claimed was more effective than the benchmark particle swarm optimization algorithm. The [25] proposed a CPSO algorithm to explore UAV routing and how it uses a computation-based optimization technique to optimize the UAV swarm performance. This algorithm is applied to solve the node localization of UAVs in space and combinational issues.

The study in [26] proposed a modified form of PSO for clustering, specifically for routing and clustering applications. Furthermore, the article explores various clustering methods for wireless sensor networks, including model-based path prediction, approximate nonlinear modeling, and feedback linearization control, and their respective advantages and disadvantages. The Multi-objective Optimization Routing Protocol [27] for UAV networks utilizes a Q-learning approach to achieve low-delay and low-energy service guarantees. While many existing Q-learning-based protocols employ fixed values for the Q-learning parameters, the proposed protocol dynamically adjusts these parameters to account for the high dynamics of FANETs. Furthermore, this paper introduces a new exploration and exploitation mechanism that enhances the discovery of potential optimal routing paths while leveraging acquired knowledge. The paper [28] proposes a hybrid routing scheme called the Position-Monitor-based Hybrid Routing Protocol (PMHRP) that takes advantage of geographic and topology-based routing protocols to address the issues of static preloaded location values. The proposed scheme offers a better packet delivery ratio, reduces the average delay, and provides more throughput than existing routing schemes.

The authors [29] introduced an optimal game routing protocol that utilizes the game payoff function based on the eigenvector connectivity value and energy parameters to select the optimal path from the source UAV to the destination. A study in [30] proposed an intelligent clustering routing approach (ICRA) for UANETs. The ICRA comprises three components: clustering, strategy adjustment, and routing modules. The reinforcement-learning-based strategy adjustment module continuously learns the benefits of different strategies for calculating the node utility. This model uses this knowledge to determine the optimal clustering strategy for the current network state. The proposed approach reduces end-to-end delay, improves packet delivery, and introduces inter-cluster forwarding nodes to forward messages between clusters. Similarly, in [31], the authors presented a new energy-aware routing scheme based on the optimized link state routing (OLSR) algorithm, which uses two processes: route discovery and route maintenance. The scheme calculates the score of a node based on its direction of movement, remaining energy, link quality, and stability. The firefly algorithm was used to select the most suitable MPR nodes.

The abovementioned schemes aim to address the challenges of UAV routing in wireless networks. However, the high mobility of UAVs and 3D environments poses new issues, such as topology-based routing schemes not maintaining a stable topology. While position-based schemes work well when a large number of UAVs or all UAVs are close to each other in the communication range, they struggle to find the optimal path if the distance between UAVs increases. To overcome these challenges, researchers have recently adopted clustering-based schemes in combination with traditional schemes. However, clustering-based schemes present their own set of challenges. One of the main challenges is that the clustering process can be complex and computationally expensive, leading to increased overhead and delay. Moreover, the performance of clustering-based schemes depends heavily on the initial phase of clustering, which can affect the overall network performance. Additionally, the use of clustering can lead to an uneven distribution of traffic and congestion in certain areas of the network, resulting in decreased throughput and increased delay. Furthermore, traditional clustering-based schemes may not be suitable for highly dynamic environments where the network topology frequently changes, as most clustering schemes form clusters in the initial phase without providing any mechanism in case UAVs exit the cluster. Table 1 provides comparative information on benchmark schemes with their strengths and weaknesses.

Table 1
Comparative table.

Routing Scheme	Strengths	Weaknesses
Hybrid bio-inspired model (PSO, GA, GWO) [22]	Minimizes distance of routing path, maximizes energy efficiency levels, and ensures fault tolerance.	Computationally expensive, may converge to suboptimal solutions or get trapped in local optima
Integrated host- and content-centric routing mechanism [23]	Harnesses advantages of both routing mechanisms, reduce flooding for path exploring, adjusts to topology dynamics	May suffer from increased latency due to route discovery and maintenance, may be vulnerable to security attacks such as node impersonation and routing loops
Particle swarm optimization algorithm [24]	Optimizes performance of UAV swarms, more effective than benchmark algorithm	Computationally expensive, may converge to suboptimal solutions or get trapped in local optima
CPSO algorithm [25]	Optimizes UAV swarm performance, suitable for node localization and combinational issues	Computationally expensive, may converge to suboptimal solutions or get trapped in local optima
Modified PSO for clustering [26]	Suitable for clustering in routing and clustering applications	May suffer from increased latency due to route discovery and maintenance, vulnerable to security attacks such as node impersonation and routing loops
Multi-objective Optimization Routing Protocol (Q-learning approach) [27]	Achieves low-delay and low-energy service guarantees, dynamically adjusts Q-learning parameters, enhances discovery of potential optimal routing paths	Computationally expensive, may converge to suboptimal solutions or get trapped in local optima
AODV [32]	Requires less overhead, handles high mobility and dynamic environments well, supports both unicast and multicast routing	May suffer from route discovery and maintenance delays in large networks, vulnerable to black hole and gray hole attacks
Intelligent clustering routing approach (ICRA) [30]	Reduces end-to-end delay, improves packet delivery, introduces inter-cluster forwarding nodes to forward messages between clusters	May suffer from increased overhead and delay due to complex and computationally expensive clustering process
Energy-aware routing scheme based on OLSR algorithm [31]	Calculates the score of a node based on its movement direction, remaining energy, link quality, and stability, uses the firefly algorithm to select most suitable MPR nodes	May suffer from increased latency due to route discovery and maintenance, may be vulnerable to security attacks such as node impersonation and routing loops

3. Methodology

The primary aim of this paper is to examine the communication routing protocol utilized by UAVs in UAV cluster applications. In these applications, drones exchange information through multiple hops, creating a decentralized network that does not rely on a central fixed node. Each drone in the swarm is equipped with a routing function that enables it to send and receive information from other drones, thereby allowing efficient communication and coordination. The ability of the swarm to function without a central controller and adapt to changing environments through local interactions among the drones makes it a promising technology for various applications, such as monitoring, surveillance, and search and rescue. Usually, wireless UAVs share data signals in free space; therefore, many factors affect the signal strength from the transmitting UAV to the receiving UAV. The attenuation factor helps determine the communication range and the required transmitted signal for efficient communication. The data or information is modulated with frequency signals as electromagnetic waves pass through free space or air. The receiving UAV antenna catches the signals and demodulates them into the original data. Due to loss in the wireless channel, the signal power received at the receiving UAV is much less than the transmitted power; the free space loss equation calculates the attenuation loss.

3.1. 3D free space

Let us consider the source UAV transmits the power through an isotropic antenna in the form of a sphere shape, and there is a point where all power is located. Hence, it is beneficial to measure the power density because the receiving antenna of the UAV can only receive a portion of the power from the transmitted power of the UAV antenna. To determine the density of an isotropic antenna, one can compute the power per square meter at varying distances.

$$\mathrm{Pd} = \frac{\mathrm{Pt}}{4\pi \mathrm{d}^2} \tag{1}$$

Here, *Pd* and *Pt* are the power density and transmitted power of the source UAV, and d is the distance from the source UAV to the destination. As we know, the Friis equation is generally considered equivalent to the power density equation; therefore, here, we have assumed.

$$\mathrm{FL} = \frac{(4\pi \mathrm{d})^2}{\mathrm{l}^2} \tag{2}$$

Here *l* is the wavelength of the signal carrier. The above equation in decibel form can be written as

$$\begin{aligned}\mathrm{FSLdB} &= [\mathrm{FSL}] = 20\log10(4\pi \mathrm{d}/\mathrm{l}) \\ &= 20\log10(4\pi) + 20\log10(\mathrm{d}) - 20\log10(\mathbf{l})\end{aligned} \tag{3}$$

There are two UAVs with isotropic antennas: one is transmitting and the other is receiving. To calculate the signal power at the receiving end, Eq. (4) describes the impact of losses on the receiving signal.

$$\mathrm{Pr} = Pt, \mathbf{FSL} = \mathbf{Pt}, ((4\pi\mathbf{d})/1)^2 = \frac{\mathbf{Pt} * \mathbf{l}^2}{(4\pi\mathbf{d})^2} \tag{4}$$

For convenience, the above equation is converted into decibel form as

$$[\mathrm{Pr}] = [\mathrm{Ps}] - [\mathrm{FSL}] = 10\log10\mathrm{Ps} - 20\log10(4\pi \mathrm{d}/\mathrm{l}) \tag{5}$$

In high-frequency transmission systems operating in the GHz range, it is important to have a direct line-of-sight between antennas. To achieve this, antenna shaping can be used to focus the transmit power towards the receive antenna, resulting in increased power density in a specific direction. This concentrated power, known as the antenna gain (G), leads to stronger signals in the desired direction compared to an isotropic antenna. As a result, directional antennas with their respective antenna gains (GT for the transmit antenna and GR for the receive antenna) are often used. The resulting link equation is expressed as follows:

$$\begin{aligned}\mathrm{Pr} &= (\mathrm{Pt} * \mathrm{GT} * \mathrm{GR}), \mathrm{FSL} \\ &= (\mathrm{Pt} * \mathrm{GT} * \mathrm{GR}), ((4\pi \mathrm{d})/\mathbf{l})2 = (\mathrm{Pt} * \mathrm{GT} * \mathrm{GR} * \mathbf{l}2)/(4\pi \mathrm{d})2\end{aligned} \tag{6}$$

Eq. (6) defines the received power (Pr) of a high-frequency transmission system using directional antennas with their respective gains (GT for the transmit antenna and GR for the receive antenna). This equation can be expressed in decibel form, as shown in Eq. (7) below.

$$\begin{aligned}[\mathrm{Pr}] &= [\mathrm{Pt} * \mathrm{GT} * \mathrm{GR}] - [\mathrm{FSL}] \\ &= 10\log10\mathrm{Ps} + 10\log10\mathrm{GT} + 10\log10\mathrm{GR} - 20\log10\left(\frac{4\pi \mathrm{d}}{\mathrm{l}}\right)\end{aligned} \tag{7}$$

The Effective Isotropic Radiated Power (EIRP) on the transmission side is often defined as the combination of the transmit power (Pt) and the transmit antenna gain (GT).

$$\mathrm{EIRP} = \mathrm{Pt} * \mathrm{GT} \tag{8}$$

We can also express the relationship between the Effective Isotropic Radiated Power (EIRP), transmit power (Pt), and transmit antenna gain (GT) in decibel form. Specifically, we have

$$[\mathrm{EIRP}] = [\mathrm{Pt}] + [\mathrm{GT}] \tag{9}$$

Substituting EIRP into Eq. (7), we will get

$$[\mathrm{Pr}] = [\mathrm{EIRP}] + [\mathrm{GR}] - [\mathrm{FSL}] \tag{10}$$

The above equation indicates the received power, which depends on the EIRP, received gain (GR), and free space loss.

3.2. Prediction-based gauss Markov 3D-mobility model

Mobility models are utilized to provide a realistic network environment and play an essential role in the validation and simulation of wireless networks. In this paper, the Gauss-Markov 3D mobility model is adopted due to its unique features, such as having memory, which gives the information nodes movement in the network. Due to the presence of memory, the Gauss-Markov 3D mobility model decreases the probability of sudden change positions of UAV in mobile ad-hoc networks as other mobility models suffer from encountering this issue (random waypoint, random walk, and random direction mobility models). Initially, each UAV is assigned a specific position, mobility, and direction every time the speed and direction of the UAVs change. In a 3D mobility model for UAV ad-hoc networks, Markov processes can describe the motion of the UAVs [33]. The Markov 3D mobility model captures the spatial and temporal correlations in the movement of UAVs. Gauss-Markov 3D mobility model, which is a stochastic model used to simulate the motion of a UAV in a 3D space Gauss-Markov 3D mobility model, used to simulate the movement of a UAV in a 3D space. The model assumes that the UAV's motion is affected by random disturbances, and the goal is to simulate the UAV's motion in a way that accurately represents the real-world behavior of the UAV. Eqs. (11), 12, and 13 are used to update the acceleration values in each direction (axe, Ay, and Az) based on the current state and a random variable sampled from a Gaussian distribution with mean 0 and standard deviation 1.

$$\mathrm{Ax} = \alpha * \mathrm{Ax}(\mathrm{t}) + (1 - \alpha) * \mathrm{X} \tag{11}$$

In Eq. (11), axe represents the acceleration in the x-direction. axe(t) is the previous value of axe, and X is a random variable obtained from a Gaussian distribution. The parameter α is a weighting factor determining the weight given to the previous value of axe (axe(t)) versus the random variable (A). Eq. (11) shows that the new value of axe is a weighted average of the previous value and a random disturbance.

$$\mathrm{Ay} = \alpha * \mathrm{Ay}(\mathrm{t}) + (1 - \alpha) * \mathrm{Y} \tag{12}$$

The variable Ay represents the acceleration in the y-direction. Ay(t) is the previous value of Ay, and Y is a random variable sampled from a Gaussian distribution. Parameter α is a weighting factor determining the weight given to the previous value of Ay (Ay(t)) versus the random variable (Y). The equation essentially states that the new value of Ay is a weighted average of the previous value and a random disturbance.

$$\mathrm{Az} = \alpha * \mathrm{Az}(\mathrm{t}) + (1 - \alpha) * \mathrm{Z} \tag{13}$$

The variable Az represents the acceleration in the z-direction. Az(t) is the previous value of Az, and Z is a random variable sampled from a Gaussian distribution. Parameter α is a weighting factor that determines how much weight is given to the previous Az (Az(t)) versus the random variable (Z). The equation essentially states that the new value of Pd is a weighted average of the previous value and a random disturbance. The original Gauss Markov 3D mobility model is designed for stationary UAVs and only predicts the current position. Therefore, to increase the accuracy of position prediction of UAVs, it is necessary to update the position, velocity, and acceleration after every time step. Hence, it is necessary to adopt position, velocity, and acceleration update equations to predict the position of each UAV at each time instant.

- Position (X, Y, Z) update at $(t + \Delta t)$

$$X(t + \Delta t) = X(t) + Vx(t) * \Delta t \quad (14)$$

$$Y(t + \Delta t) = Y(t) + Vy(t) * \Delta t \quad (15)$$

$$Z(t + \Delta t) = Z(t) + Vz(t) * \Delta t \quad (16)$$

where `X(t)`, `Y(t)`, and `Z(t)` are the UAV's X, Y, and Z coordinates at time `t`, respectively; `Vx(t)`, `Vy(t)`, and `Vz(t)` are the UAV's X, Y, and Z velocities at time `t`, respectively; and `Δt` is the time step. Based on the position updates of the 3D Cartesian coordinates. The pitch angle of each UAV in the free space can be calculated from Eq. (17)

$$\theta = \mathrm{atan2}\left(-y, \sqrt{x^2 + z^2}\right) \quad (17)$$

Here, the atan2 function is used with the negative y-coordinate and the square root of the sum of the squared x and z coordinates. This equation considers the y-axis's orientation in a right-handed coordinate system, where the y-axis is typically oriented downwards. The resulting value is the pitch angle of the UAV in radians, which is an important parameter in controlling the UAV's flight and stability

- Velocity ($\mathbf{V_X}$,$\mathbf{V_Y}$,$\mathbf{V_Z}$) update at $(t + \Delta t)$

$$Vx(t + \Delta t) = Vx(t) + Ax(t) * \Delta t \quad (18)$$

$$Vy(t + \Delta t) = Vy(t) + Ay(t) * \Delta t \quad (19)$$

$$\mathbf{Vz(t + \Delta t) = Vz(t) + Az(t) * \Delta t} \quad (20)$$

where `axe(t)`, `Ay(t)`, and `Az(t)` are the UAV's X, Y, and Z accelerations at a time `t`, respectively.

- Acceleration (axe, Ay, Az) update at $(t + \Delta t)$

In order to integrate the Markov process, accelerations are updated considering both the updated state at t + Δt time instant and a random variable. The subsequent equations provide the necessary framework.

$$Ax(t + \Delta t) = \alpha * Ax(t) + (1 - \alpha) * \sigma x * R(t) \quad (21)$$

$$Ay(t + \Delta t) = \alpha * Ay(t) + (1 - \alpha) * \sigma y * S(t) \quad (22)$$

$$Az(t + \Delta t) = \alpha * Az(t) + (1 - \alpha) * \sigma z * T(t) \quad (23)$$

where `α` is the memory factor (between 0 and 1) that determines the correlation between the current and next acceleration; `σx`, `σy`, and `σz` are the standard deviations of the X, Y, and Z accelerations, respectively; and `R(t)`, `S(t)`, and `T(t)` are random variables sampled from a Gaussian distribution with mean 0 and standard deviation 1. In order to predict the next location of UAVs, the Markov mobility equations for UAV ad-hoc network is expressed as follows:

$$P(X(t + \Delta t)|X(t)) = \Sigma\, P(X(t + \Delta t)|X(t - 1)) * P(X(t - 1)|X(t - 2)) * \ldots * P(X(1)) \quad (24)$$

Where:

- $P(X(t+\Delta t)|X(t))$ is the probability of the UAVs being in a particular location and velocity at time $t+\Delta t$, given their current location and velocity at time t.

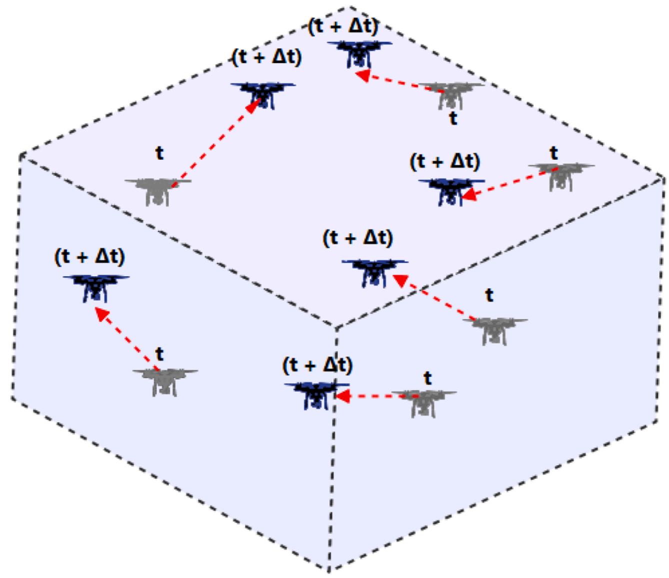

Fig. 1. Prediction-based Gauss Markov 3D-mobility model.

- $P(X(t+\Delta t)|X(t-1))$ is the probability of the UAVs being in a particular location and velocity at time $t+\Delta t$, given their location and velocity at time t-1.
- $P(X(t-1)|X(t-2))$ is the probability of the UAVs being in a particular location and velocity at time t-1, given their location and velocity at time t-2.
- $P(X(1))$ is the initial probability distribution of the location and velocity of UAVs at time $t = 1$.

To solve the Markov mobility equations, it is necessary to define the transition probabilities $P(X(t+\Delta t)|X(t-1))$. These probabilities are calculated based on the UAVs' movement patterns and the dynamics of the environment in which they operate; hence, the transition probabilities are as follows:

$$P(X(t + \Delta t)|X(t - 1)) = N(X(t), \sigma) * T(X(t), \Delta t) \quad (25)$$

In the above Equation N(X(t), σ) is a Gaussian distribution that represents the random movements of UAVs around their current location X(t), with standard deviation σ. T(X(t), Δt) is the transition matrix that represents the UAVs' motion over a time interval Δt, based on their current location X(t) and their speed and direction of movement. The transition matrix T(X(t), Δt) can be calculated using the UAVs' kinematic equations of motion, such as the velocity, acceleration, and position update equations described above. These equations consider the UAVs' speed, the direction of movement, and any external forces acting on them, such as wind or turbulence. The UAVs are moving in a three-dimensional coverage area; therefore, it is necessary to consider the movement of each UAV in the network (Fig. 1).

3.3. K-Means online-learning routing protocol (K-MORP)

This subsection presents two main phases of the proposed routing algorithm based on K-MEAN online learning. In the first phase, the mathematical model is developed, while in the second phase, the proposed algorithms are formulated. Basically, these algorithms were implemented in network simulator three for simulation purposes.

3.3.1. Constrained-Based k-Means online learning clustering

Let's assume there are $N = \{u_1, u_2, u_3 \ldots .. u_n\}$ number of UAVs in the wireless network to have efficient routing and data forwarding, the whole network is divided into different clusters. To form clusters in the wireless network, initially k means clustering technique is adopted. There are $K = \{k_1, k_2, k_3 \ldots .. k_m\}$ number of clusters in the having $\mu_K = \{\mu_1, \mu_{2,}, \ldots . \mu_m\}$*centriods*. The position of each UAV is determined by applying Euclidean distance Eq. (26), and centroids are ascertained using the mean formula as shown in Eq. (27).

$$|U_j| = \sqrt{(ux_j - ux_i)^2 + (uy_j - uy_i)^2 + (uz_j - uz_i)^2} \tag{26}$$

where U_j is any UAV in the wireless ad-hoc network having ux_j,uy_j and uz_j are its 3D coordinates relative to the point coordinates ux_i,$uy_i$and uz_i.

$$\mu_m = \left[\left(\frac{ux_1 + ux_2 + \ldots + ux_n}{m}\right), \left(\frac{uy_1 + uy_2 + \ldots + uy_n}{m}\right), \left(\frac{uz_1 + uz_2 + \ldots + uz_n}{m}\right)\right] \tag{27}$$

Based on the mean distance and positions of UAVs, the coverage area of the whole network is calculate by considering the 3D coverage as the volume V (C). Assuming that distance (U_j, μ_j) is the distance between each UAV U_j and the centroid μ_j . As the UAVs are moving in 3D coverage area therefore the coverage distance is considered as 3D volume as the sum of the volumes covered by each UAV can be evaluated as

$$V(C) = \sum_{k=1}^{m} \sum_{n=1}^{N} \text{Volume of coverage } (U_j, \mu_j) \tag{28}$$

In the above equation V (N) is total 3D coverage volume of whole network, K is the total number of centroid and N is total Number of UAVs. Moreover, to calculate the VolumeofCoverage(U_j,μ_j) assuming it as rectangular box or prism having 3D coordinates x, y and z. Initially, 3D volume is calculated by multiplying length (l), height (h) and width (w) as described below.

$$Volume = l \times w \times h \tag{29}$$

To calculate the coverage volume using Cartesian coordinates, it is necessary to determine the distances in the x, y, and z directions between the UAV node and the centroid. Let us denote these distances as the dx, dy, and dz directions, respectively. Assuming the UAV U_n and the centroid μ_k are represented by their Cartesian coordinates (uj_x,uj_y,uj_z) and (μj_x, μj_y, μj_z), the lengths in each direction can be calculated as:

$$l = |uj_x - \mu j_x| + dx \tag{30}$$

$$w = |uj_y - \mu j_y| + dy \tag{31}$$

$$h = |uj_z - \mu j_z| + dz \tag{32}$$

Here, the absolute value is used to ensure positive lengths. By substituting these dimensions into the volume formula, the coverage volume influenced (VoI) by the distance between the UAV node and the centroid can be calculated as

$$\mathrm{VoI}(U_j, \mu_j) = (|uj_x - \mu j_x| + dx) \times (|uj_y - \mu j_y| + dy) \times (|uj_z - \mu j_z| + dz) \tag{33}$$

The above equation calculates the rectangular cuboid volume based on the distances in each direction. The dimensions dx, dy, and dz represent the influence or coverage range in each axis based on the distance between the UAV node and centroid. After calculating the coverage area and centroid of the cluster next step is to find the distance difference between the μ_j centroid and given point u_i is called a mean square (MSE).

$$MSE = \sum_{j=1}^{m} \sum_{i=1}^{n} |\ u_i - \mu_j|^2 \tag{34}$$

The above equation can also be written as

$$MSE = \sum_{j=1}^{m} \sum_{i=1}^{n} u_{i,j}|\ u_i - \mu_j|^2 \tag{35}$$

In UAV networks, the main goal is to minimize the MSSE to reach the optimal value to form clusters. Therefore, differentiation of MSSE has been performed with respect to two factors. One is μ_j the centroid position and other is each UAV position. $u_{i,j}$. Initially, take μ_j is constant and $u_{i,j}$ will be obtained.

$$\frac{\Delta MSSE}{\Delta u_{i,j}} = \sum_{i=1}^{n} \sum_{j=1}^{m} u_i - \mu_j^2 \tag{36}$$

If the distance between $u_i and \mu_j$ is minimum, there will be less MSSE can be evaluated as

$$UAV(u_{i,j}) = \mathrm{argmin} u_i - \mu_j^2 \tag{37}$$

Based on the aforementioned equation, it is evident that

$$u_{i,j} = \begin{cases} 1 & \text{if } u_{i,j} \text{ belongs to k cluster} \\ 0 & \text{otherwise} \end{cases} \tag{38}$$

Keep $u_i$is constant and μ_j is varying, we will get

$$\frac{\Delta MSSE}{\Delta \mu_k} = 2 * \sum_{j=1}^{m} u_i u_i - \ \mu_j \tag{39}$$

By solving the above equation with respect to μ_j

$$C_j = \frac{\sum_i^n \mu_j * u_i}{\sum_i^n u_i} \tag{40}$$

Here C_j is jth cluter having μ_j centroid point its can be calculated from Eq. (31) size and its UAVs. However, forming clusters in the UAVs scenario is challenging due to the presence of similar UAVs in different clusters, the size of each cluster, and the iteration process. To overcome these issues, a constrained-based k-means clustering scheme is introduced to avoid the overlapping issue and many iterations.

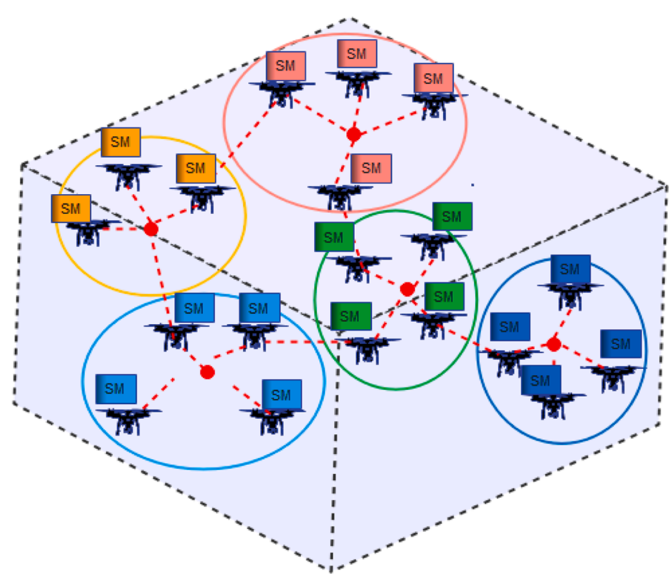

Fig. 2. Constrained-based k-means online learning clustering for UAV Ad-hoc network.

$$\mathrm{MSE} = \sum_{j=1}^{m}\sum_{i} u_{i,j}\left| u_i - \mu_j\right|^2 \qquad \forall \sum_{j=1}^{K}\mu_j \sum_{i}^{n} u_i < \tau \tag{41}$$

Where $\tau$is the upper bound, which is the size of the cluster and the number of iterations; further, we have introduced a two-similarity matrix. S_c and C_A The first one shows the similarity score inside the clusters, while the other shows the number of iterations to form a cluster. The online cluster constraint function is described as follows:

$$O_c = \sum_{c}\sum_{A}(S_c\ ,C_A\ )\forall A \sum_{A=1}^{A=n} S_c \geq S_\gamma \tag{42}$$

The above EquationO_c is an online constraint for the formation of clusters, S_c and C_A Shows the similarity and iteration information. In the above equation, the iteration number is set for a specific time; therefore $S_c \geq S_\gamma$ must follow this condition. Fig. 2 shows the network architecture of the K-MORP, in which the UAVs are distributed in different clusters, as shown in C1, C2, C3, C4, and C5.

3.3.2. Algorithm-based description of K-MORP

These algorithms are interconnected to select optimized clusters based on multiple techniques; therefore, for simplicity and understanding, we distribute our proposed work into multiple algorithms.

A. K-means clustering for UAV placement optimization. K-means Clustering is a well-known unsupervised learning algorithm that partitions a dataset into K clusters based on the similarity of data points. It aims to group similar UAVs while minimizing the intra-cluster distance and maximizing the inter-cluster distance. Initially, Algorithm 1 randomly selects K data points from the dataset as the initial centroids for the K clusters at line 1. Line 2 describes that each UAV is assigned to the closest centroid based on the Euclidean distance Eq. (26) for measuring the distance between the data point and the centroid. The commonly used distance metric is the Euclidean distance, which calculates the straight-line distance between two points in n-dimensional space. After assigning data points to centroids, the position of each centroid is updated to the mean position of the data points in its cluster, computed using (27) at line number 3. Line 4 calculates the mean square error (MSE) using (34), and line 5 maintains the updated positions of UAVs with the help of algortihm2. The algorithm repeats this process until the centroids converge and stop moving significantly or reach the maximum number of iterations (from lines 6–7).

The primary goal of Algorithm 1 is to reduce the within-cluster sum of squares, which is the total of the squared distances between all data points and their respective centroids. This metric is known as the objective function, and the algorithm attempts to identify the best clustering solution that minimizes this function. However, K-means Clustering has a drawback in that the initial placement of centroids can significantly impact its performance. If the initial centroids are not well positioned, the algorithm may converge to a suboptimal clustering solution.

Algorithm 1

K-means clustering for UAV placement optimization.

Input: N: number of UAVs; K: number of clusters; UAVs: list of UAVs with positions; distance: distance metric for calculating distance between UAVs and cluster centroids (e.g., Eq. (26))

Output: A list of K optimal cluster centroids representing the best locations for placing UAVs

1. centroids = randomly select K UAVs from UAVs // Initialize random centroids
2. clusters = assign UAVs to nearest centroid using (26)
3 centroids = update centroids to mean of UAV positions in each cluster using (27)
4. mse = calculate mean square error for each cluster (34)
5. updated_uav_positions = optimize clustering using Algorithm 2
6. if convergence_criteria_met: // Check for convergence break;
7 return centroids // Return list of K optimal cluster centroids

Algorithm 2

A modified k-means clustering algorithm with constrained-based optimization.

Input: learningRate: learning rate for updating centroids; clusters: list of K clusters; centroids: list of K centroids; UAVs: list of UAVs with positions

Output: The list of UAVs with updated positions

1. // Optimize clustering by updating centroids and UAV positions
2. for $i = 1$ to K:
3. // Calculate partial derivative of MSE with respect to centroid
4. ∂MSE(dc) = empty point
5. for each UAV u in cluster[i]:
6. d = distance(u.position, centroids[i])
7. $\frac{\partial MSE}{\partial dcx}$+= (d - centroids[$i$].x) × (1/size of cluster[i])
8. $\frac{\partial MSE}{\partial dcy}$ += (d - centroids[i].y) × (1/size of cluster[i])
9. $\frac{\partial MSE}{\partial dcz}$ += (d - centroids[i].z) × (1/size of cluster[i])
10. // Calculate partial derivative of MSE with respect to UAV position
11. for each UAV u in cluster[i]:
12. ∂MSE(ui) = empty point
13. d = distance(u.position, centroids[i])
14. $\frac{\partial MSE}{\partial ux}$ = (u.position.x - centroids[i].x) × (2/size of cluster[i])
15. $\frac{\partial MSE}{\partial uy}$= (u.position.y - centroids[i].y) × (2/size of cluster[i])
16. $\frac{\partial MSE}{\partial uz}$ = (u.position.z - centroids[i].z) × (2/size of cluster[i])
17. // Update UAV position
18. new_position = empty point
19. for $j = 1$ to K:
20. dist_to_centroid = distance(u.position, centroids[j])
21. if dist_to_centroid < distance(u.position, centroids[i]):
22. new_position += learningRate × $\frac{\partial MSE}{\partial ui}$ × (1/dist_to_centroid)
23. // Call Algorithm 3 to check if the cluster meets the constraint
24. if size of clusters[j] < τ and Algorithm_3(cluster[i], centroids[i], UAVs, S_γ) == False:
25. assigned = False
26. Break
27. else:
28. assigned = True
29. *new_cluster* = j
30. if assigned == True:
31. remove u from cluster[i]
32. add u to cluster[new_cluster]
33. centroids[i] = mean(cluster[i])
34. // Return selected clusters that meet constraint
35. selected_clusters = empty list of K clusters
36. for $i = 1$ to K:
37. add clusters[i] to selected_clusters
38. return selected_clusters

B. a modified k-means clustering algorithm with constrained-based optimization. The pseudo-code of Algorithm 2 describes an iterative algorithm that optimizes k-means clustering under size constraints on clusters.

It does this by updating the positions of the cluster centroids and UAVs in a constrained manner. For each k cluster, the algorithm first calculates the partial derivative of the mean squared error (MSE) with respect to the centroid position (lines 4–9). This calculation determines how much the MSE would decrease by moving the centroid by a small amount in each dimension. Next, the algorithm calculates the partial derivative of the MSE for each UAV's position, which determines how much the MSE would decrease by moving the UAV by a small amount (lines 11–16). Using these derivatives and a preset learning rate, the algorithm updates the position of each UAV to decrease the MSE. However, it also checks whether reassigning the UAV to the new closest centroid violates the size constraint on that cluster. If the reassignment does not violate the constraint, the UAV is reassigned to that cluster; otherwise, it remains in its original cluster (lines 18–30). Additionally, in line 23, the algorithm calls Algorithm 3 the constraint function to ensure that the clustering meets the size constraint. Once all UAVs in a cluster have been considered, the centroid of that cluster is recalculated as the mean of the UAV positions to incorporate the changes (line 33).

Algorithm 3
Pseudo code of an online cluster constraint function.

```
Input: - S_γ: similarity score threshold; - clusters: list of K clusters; - centroids: list of K
  centroids; - UAVs: list of UAVs with positions
  Output: selected_clusters list
1.  // Apply online cluster constraint function
2.  for i = 1 to K:
3.  similarity_score = 0
4.  num_iterations = 0
5.  for each pair of UAVs (u1, u2) in cluster[i]:
6.  similarity_score += similarity(u1, u2)
7.  num_iterations = clustering_function(cluster[i])
8.  if similarity_score + num_iterations < S_γ:
9.  // Reassign UAVs to the nearest centroid and recalculate centroids
10. for each UAV u in cluster[i]:
11. min_dist = infinity
12. nearest_centroid = None
13. for each centroid c in centroids:
14. d = distance(u.position, c)
15. if d < min_dist:
16. min_dist = d
17. nearest_centroid = c
18. remove u from cluster[i]
19. add u to the cluster associated with nearest_centroid
20. centroids[i] = mean(clusters[i])
21. // Return selected clusters that meet the constraint
22. selected_clusters = empty list of K clusters
23. for i = 1 to K:
24. add clusters[i] to selected_clusters
25. return selected_clusters
```

This iterative process of calculating derivatives, updating positions in a constrained manner, and recalculating centroids is repeated for a fixed number of iterations or until convergence occurs.

The goal of this modified k-means clustering algorithm is to minimize the MSE of the clustering under size constraints on the clusters. Finally, the algorithm returns the final set of k clusters that satisfy the size constraints (lines 35–38). This modified k-means algorithm improves upon the original k-means algorithm by enforcing constraints on cluster sizes while optimizing an objective such as MSE.

C. online cluster constraint function. This function applies an online cluster constraint by iterating over each cluster i in the list of K clusters. For each cluster (i), the function calculates the similarity score between each pair of UAVs in the cluster and the number of iterations required for clustering (lines 5–7). The similarity score is calculated by taking the sum of the similarity values between each pair of UAVs in the cluster. The number of iterations required for clustering is obtained by calling a clustering function on the current cluster i (line 7). The similarity score and number of iterations are then added (line 8), and if the sum is less than the similarity score threshold, the function applies a reassignment step. The reassignment step involves reassigning each UAV in the cluster to the nearest centroid and recalculating the centroids. The nearest centroid is obtained by calculating the Euclidean distance between the UAV position and each centroid in the list of K centroids (lines 13–17). The UAV is then removed from the current cluster (i) and added to the cluster associated with the nearest centroid (lines 18–19). The new centroid of the cluster is calculated as the mean of the new cluster (line 20). Finally, the function returns a list of selected clusters that satisfy the constraint. This is achieved by adding each cluster that meets the constraint to a list of selected clusters (lines 23–24). The list of selected clusters is then returned as the output of the function (line 25).

3.4. Working principle

In this sub-section, the working principle of the proposed scheme forms clusters of the whole network in four main steps; each step is described below.

Step 1: Cluster Formation

To form clusters in the UAV network, the k-means clustering algorithm is used. Initially, the entire network is divided into K clusters. The centroids of these clusters are calculated using the mean formula from Eq. (27). The position of each UAV is calculated based on the Euclidean distance from the centroids using Eq. (26). This step results in the formation of clusters of UAVs that are geographically close to each other. The coverage area of the entire network is calculated by summing the distances between each UAV and its nearest centroid for all clusters, as shown in Eq. (28). This step helps determine each cluster's geographical coverage and the entire network. Fig. 3(a) shows that in the initial phase at a time interval (t) randomly, C1 and C2 clusters are formed. The red color UAVs are cluster head UAVs in each cluster, and the Euclidean formula calculates the distance between each UAV.

Step 2: Mean Square Error (MSE) Calculation

The mean square error (MSE) is calculated to find the distance difference between the centroids and each UAV position using Eqs. (29) or (30). This step helps to evaluate the quality of the formed clusters by measuring the distance between each UAV and its assigned centroid. Initially, the centroids are taken as constant, and the UAV's change position is calculated as shown in Fig. 3(b). Based on the new positions, the updated centroid is also estimated respectively. Then, UAVs are

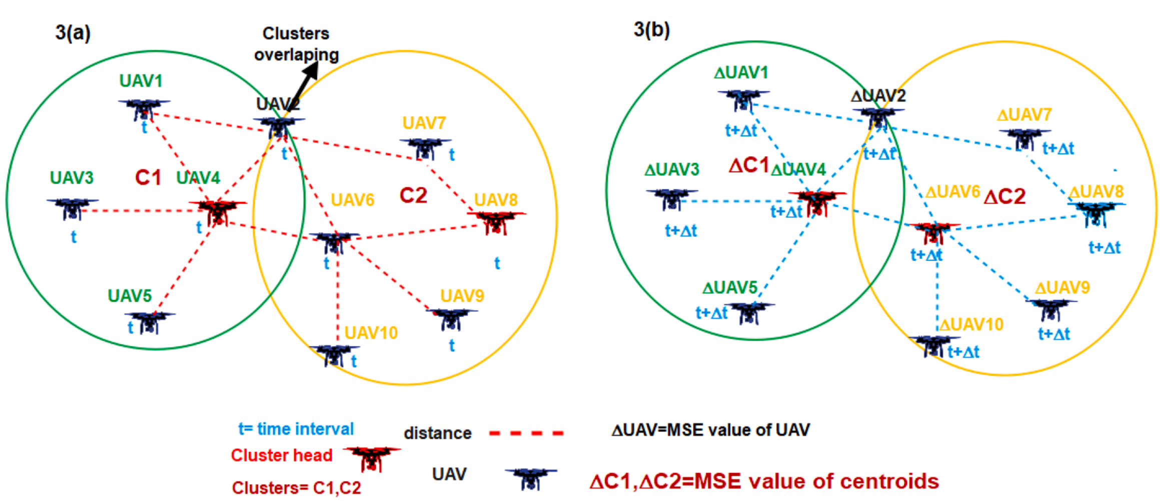

Fig. 3. (a) Cluster formation; 3(b) MSE calculation of the UAV Ad-hoc Networks.

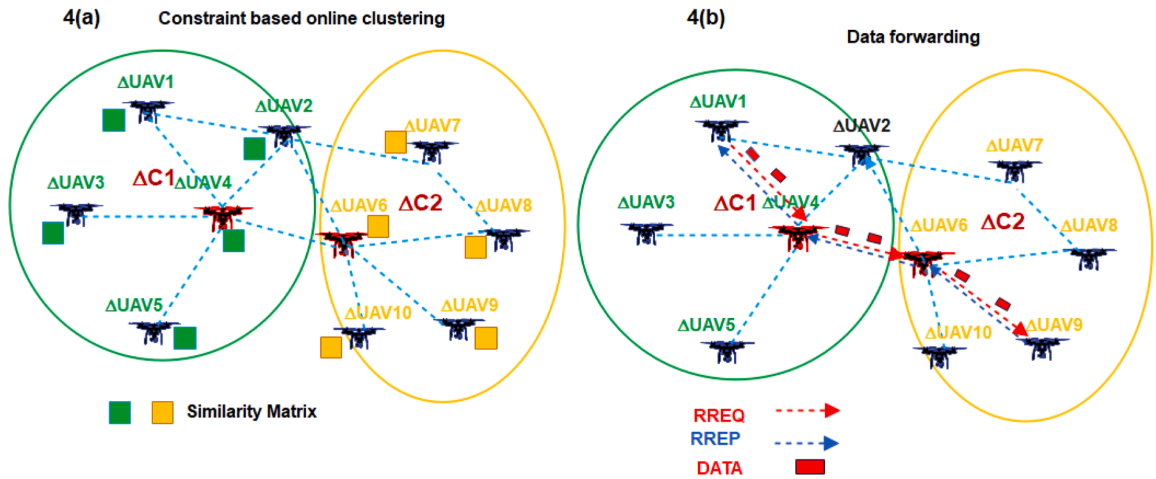

Fig. 4. (a) Constrained-Based online clustering; 4(b) Data forwarding of the UAV Ad-hoc Networks.

assigned to the nearest cluster based on the minimum distance between the UAV and the centroid of each cluster using Eq. (33). This step assists in identifying the membership of each UAV in a specific cluster.

Step 3: Constrained-Based k-Means Clustering

To address overlapping issues and iterations, constrained-based k-means clustering is introduced, which restricts the size of each cluster and the number of iterations using Eq. (36). This step helps improve the efficiency of the clustering process by limiting the size and number of iterations required to form each cluster. Fig. 4(a) shows that UAVs are divided into different clusters based on the SM similarity matrix(SM); UAVs with similar SM values are grouped in the same cluster.

Given the mobility of UAVs, their position changes over time, making it imperative to reinitiate the clustering process should UAVs move out of their cluster range. To tackle this issue, constraint-based k-means clustering adopts a similarity matrix, which is a combination of cluster size based on the distance of UAVs with respect to cluster head and time iteration or time interval (time interval is counted based on Prediction-based Gauss Markov 3D-Mobility Model). If a UAV moves beyond a predefined distance threshold from its assigned centroid, the re-clustering process is triggered.

Step 4: Data forwarding

After the formation of clusters, source UAV1 inquires about the route by sending an RREQ message to its cluster head. The cluster first searches if the destination node belongs to the same cluster or another cluster; in case the destination UAV belongs to another cluster, the cluster head of the C1 cluster forwards the RREQ message to the cluster head of the C2 cluster to which the destination UAV belongs, as shown in figure 4(b). The cluster head of C2 sends the RREQ request to the destination UAV. In return, the RREP is forwarded by the destination UAV by following the same process. Finally, after obtaining the RREP, the source UAV forwards the data to the destination through the cluster heads of both clusters.

If a disconnection is detected, the new cluster head initiates the reattempt of path discovery by sending an RREQ message containing information about the previous cluster head or broken link. This RREQ message is forwarded to the new cluster head, ensuring that the path can be reestablished. In addition, in the case of re-clustering events, the new cluster head updates its routing table and notifies the affected nodes to reinitialize the path discovery process. This mechanism ensures that the nodes are aware of the re-clustering event and can establish new paths based on the updated cluster structure. Finally, upon receiving the RREP, the source UAV forwards the data to the destination through the cluster heads of both clusters. Detailed data forwarding process is shown in Fig. 5.

4. Simulation and performance analysis

In this section, we compare the K-MEAN Online Learning Routing Protocol (K-MORP) proposed in this paper with benchmark routing protocols, including K-MEANs [34], AODV [32], OLSR [35], SPO [25], CSPO [26], IHCR [23], and MDAC [36]. To evaluate the performance of these routing protocols, Network Simulator Three (NS-3.35) [32], an event-driven wireless network simulator, is used that allows for accurate and efficient simulation of wireless networks. Specifically, we compare the performance of K-MORP with respect to various metrics, such as packet delivery ratio, end-to-end delay, network throughput, normalized routing load, and mean square error (MSE), against the baseline protocols. This comparison will help us determine the effectiveness of K-MORP in improving the performance of UAV swarm networks.

4.1. Simulation environment

The simulations were conducted using the NS-3 network simulator [37] running on a Core i5 with a 16GB RAM machine. The simulation environment consisted of six different scenarios with different UAV nodes having the ability of ad hoc communication, placed within a 3D box-shaped area, and implemented with the Gauss-Markov mobility model. The nodes were equipped with IEEE 802.11ac wireless network interface cards and used the ad hoc network mode of IEEE 802.11ac. The transmission power was set to 20 dBm. The simulation was run for 120 s. Table 2 lists the simulation parameters adopted during the simulation, maintaining the same parameters for all routing protocols. Table 3 lists the additional parameters used to set up the Gauss Markov Mobility Model in the NS3 simulator.

4.2. Analysis of results

This section presents the results of our simulation study, comparing the performance of the proposed KMORP with that of existing wireless ad hoc network routing protocols. To evaluate KMORP's effectiveness of KMORP using metrics such as Mean Square Error, PDR, End-to-End Delay, Throughput, and NRL. By comparing these metrics, we highlight the strengths and weaknesses of the KMORP and provide recommendations for improving network performance. Overall, our results demonstrate the potential of the KMORP to improve the performance of wireless ad hoc networks.

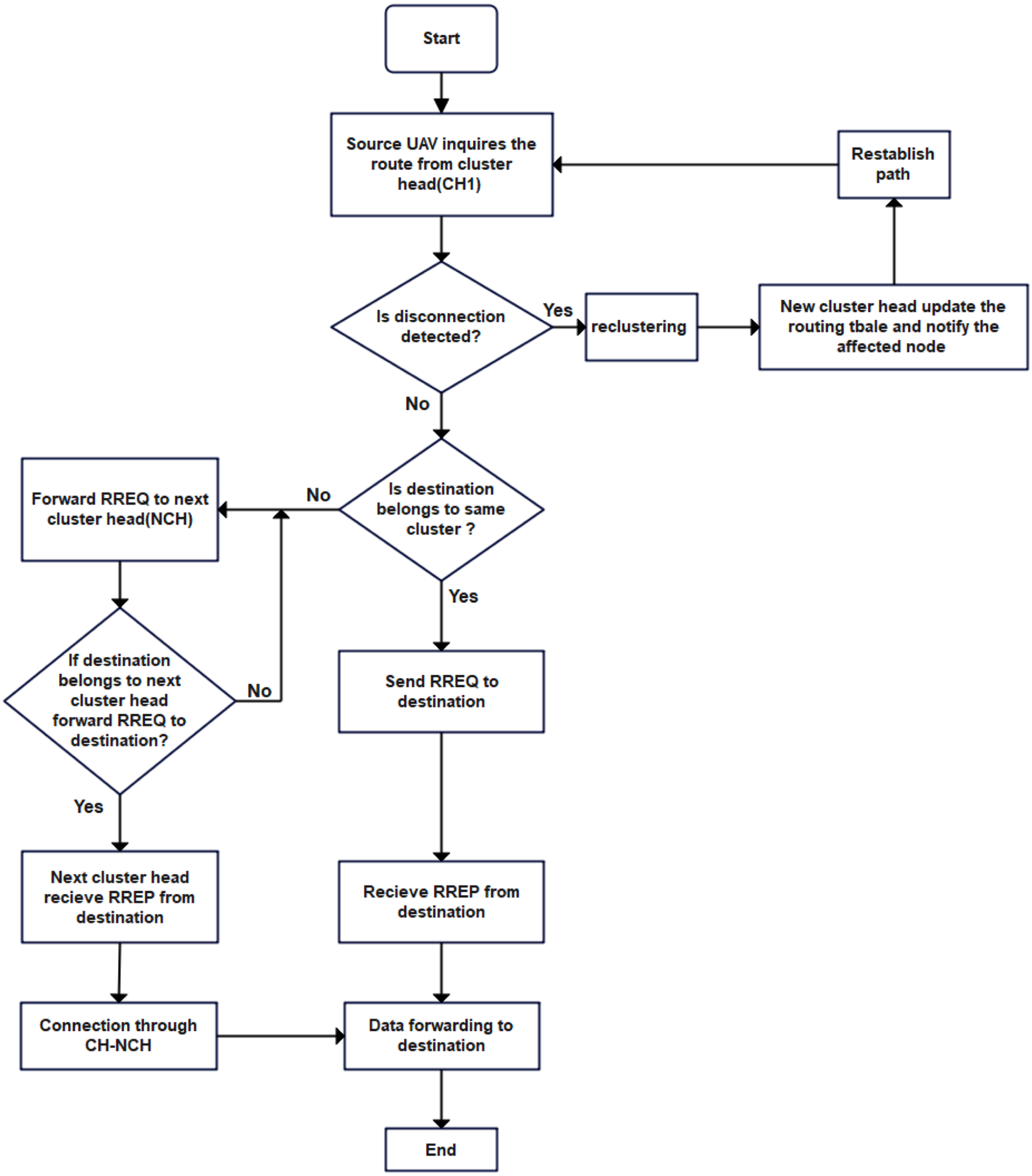

Fig. 5. Data forwarding flowchart.

A. mean square error

The mean square error (MSE) in wireless networks is used to measure the accuracy of routing protocols in predicting the routes taken by data packets from the source UAV to the destination UAV. Fig. 6 presents the results of MSE for the four routing protocols, including K-MORP, K-means, CPSO, and PSO, at different network sizes (15, 30, 45, 60, 75, and 90 nodes). The results showed that the MSE values varied for each routing protocol and network size. The figure clearly shows that K_MORP achieved the lowest MSE value of 0.05 at 15 nodes, indicating that it was the most accurate routing protocol for smaller network sizes. K-MORP was the most consistent and precise routing protocol at all network sizes, with low MSE values ranging from 0.05 to 0.23, making it the best routing protocol for UAV ad-hoc networks. While K-Means offers the second lowest MSE values ranging from 0.6 to 1.25. SPO had the highest MSE values at all network sizes, ranging from 0.9 to 1.6, indicating that it was the least accurate routing protocol. CSPO had mixed performance, with relatively low MSE values at smaller network sizes but higher MSE values at larger network sizes. The higher MSE value indicates that the other schemes suffer from the formation of optimal clusters because they fact these schemes do not operate on real-time UAVs position updates. Another reason is that clusters are formed in the initial phase; there is no mechanism in case a UAV goes outside of the cluster at any time instant. The proposed scheme K-MORP fulfills these two gaps by adopting the Gauss Markov mobility 3D model, which gives the current position of each UAV as well as predicts the future position based on velocity, acceleration, and position update as key parameters. Moreover, K-MORP operates online constraint-based clustering when the position of UAVs changes and the process of clustering formation is reinitiated. Hence, the MSE values of K-MORP are much lower than those of the benchmark schemes.

Table 2
Simulation parameters.

Simulation Parameter	Value
NS-3 Version	3.35
Computer Hardware	Ubuntu 22.04 LTS in Core i5 12 G 16GRAM, GPU 3050Ti.
Number of **nodes**	15, 30, 45, 60, 75 and 90
Simulation area dimensions	1000 × 1000 × 80 (m^3)
Wireless Network Type	IEEE 802.11ac Ad hoc mode(Wi-Fi 5 g)
Central frequency	5Ghz
Link Speed/Data rate	6.5Mbps
Channel width	20 Mhz
number of spatial streams (NSS)	1
Source : destination UAVs	3:3
Offered data rate per source	117.76 Kbps
Propagation Delay	Constant Speed Propagation Delay Model
PropagationLoss	Friis Propagation Loss Model
Transmission Power	20 dBm
Simulation Duration	120 s
Routing Protocol	K-MORP, K-MEANs, IHCR, MDAC, AODV, OLSR, SPO, and CSPO
UAV average speed	15 m/s
UDP Packet Size	1472 Bytes
Mobility Model	Gauss-Markov

Table 3
Parameters for Gauss Markov Mobility Model.

Parameters	Value
Area	X[*min*=10 *max*=1000]; Y[*min*=10 *max*=1000]; Z[*min*=5 X*max*=80];
TimeStep	5 S
Alpha	0.85
MeanVelocity	Min=0, Max=20
MeanDirection	Min=0, Max=6.283185307
MeanPitch	Min=0.05, Max=0.05
NormalDirection	Mean=0.0, Variance=0.2,Bound=0.4
NormalPitch	Mean=0.0, Variance=0.02, Bound=0.04

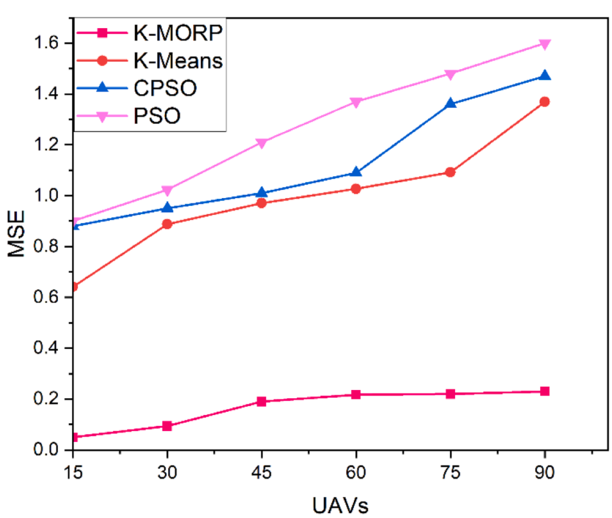

Fig. 6. Mean Square Error (MSE) versus the Number of UAVs.

B. packet delivery ratio

The packet delivery ratio is an important metric to describe the performance of the UAV routing protocol, which describes the number of packets delivered divided by the total number of packets forwarded. Fig. 7 clearly shows that the proposed K-MORP scheme offers better results, ranging from 94.21 −85.6. Although the number of UAVs increases, the proposed scheme sustains its delivery ratio, while the PDR of other routing protocols decreases as the number of UAVs increases; the delivery ratio of all routing protocols is below 80% at the highest number of UAVs (90). The reason behind the decrement in the ratio is the complicated Gauss Markov mobility model, which considers three main parameters: position, velocity, and acceleration of UAVs at each time interval, making it difficult to predict the path from source UAV to destination UAV. On the other hand, the proposed scheme adopted the online clustering mechanism with constrained based efficiently predicting the position of each and forming optimal clusters, making it appropriate to find the optimal path from the source UAV to the destination UAV. Furthermore, we can observe from the figure that the traditional k-means clustering routing scheme, which forms a cluster, makes it easy to find a path from UAV clusters in the network. However, the traditional k-means clustering routing scheme suffers due to clusters overlapping, and some UAVs remain outside of clusters, which makes it necessitates to find the optimal path. The IHCR provides the second highest results after the proposed scheme, which offers good results on 2D coverage areas and adopts a random waypoint mobility model. However, this model is not appropriate for 3D space. Initially, IHCR PDR % starts from 90%; however, when the number of UAVs increases, its PDR decreases and is slightly higher than the traditional k-means and MDAC PDR at the highest number of UAVs. One factor is that we have adopted Gauss Markov 3D mobility, which is a complex model that considers the pitch angle and other parameters that affect the performance of IHCR. While MDAC offers the third-best results, this scheme also works based on traditional k-means learning; therefore, the results are almost similar, although there is a slight improvement due to the auto-encoder. Furthermore, the proposed scheme offers more than 10% better results in comparison with IHCR and 12.96% better results than MDAC. The other benchmark schemes suffer from forming optimal paths due to the high mobility of UAVs, making link breakage very often and, therefore, have unstable paths. Fig.7 clearly shows that when the number of UAVs increases, the traditional topology-based schemes AODV and OLSR PDR% drastically decrease and reache 55% at 90 UAVs. In fact, the higher number of UAVs also makes the network topology unstable, and routes across the network changes dynamically; as a result, the topology-based schemes suffer to find the optimal path. In comparison with AODV and OLSR, the proposed scheme, K-MORP, offers 34.69% and 38%. On the other hand, PSO and CPSO can be stuck in local optima, which leads the algorithms to converge to a suboptimal solution instead of the global optimal solution. In the case of UAV networks, this can result in lower PDR values because the PSO algorithm may not be able to find the optimal routing paths.

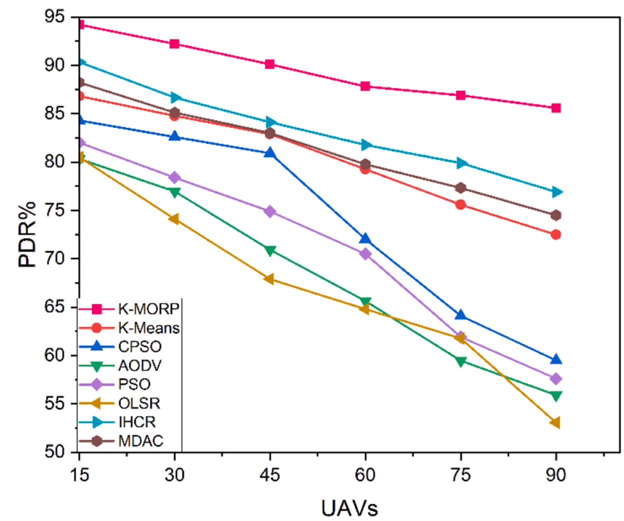

Fig. 7. Packet Delivery Ratio (PDR%) versus the Number of UAVs.

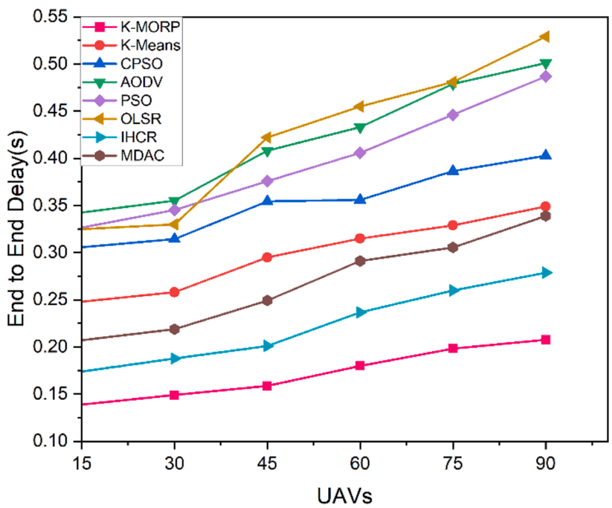

Fig. 8. End-to-End Delay(s) versus the number of UAVs.

C. end-to-end delay

The end-to-end delay measures the time a packet requires to travel from the source UAV to the destination. It stands as a primary metric of the routing protocols' performance within the network; a lower delay value means the routing protocol is more efficient. The figure shows the delay value versus the number of UAVs (15, 30,45,60,75, and 90) moving in the 3D network environment. From the figure, it can be observed that the delay value of the proposed scheme K-MORP at the initial point when the number of UAVs is 15 is 0.18, and at the highest number of UAVs still, the delay of the proposed scheme is much less than other schemes. Whereas the delay of the other schemes is higher than that of K-MORP, even the IHCR clustering delay starts from 0.19 and reaches 0.29, although it offers the second-best results in the UAV network environment. The third scheme provides better results. The other scheme, MDAC, which offers a third less delay, ranges from 0.19 to 0.348 due to the use of an autoencoder. There is an improvement in the results as compared to traditional k-means clustering. While MDAC delay increases as the number of UAV mobility increases due to the fact it operates on an OLSR routing mechanism, it also suffers in establishing the paths efficiently; as a result, its delay increases.

On the other hand, other schemes' delay value increases quickly due to the high movement of UAVs. The communication link between UAVs is unstable, which leads to higher delay values. For benchmark topology-based schemes like AODV and OLSR, the delays (s) are in the range from 0.33 to 0.51 and from 0.32 to 0.52, respectively. Moreover, SPO and CSPO offer 0.29 and 0.40, respectively, at 90 UAVs, which is a higher value than that of our proposed scheme. The proposed scheme K-MORP adopted an online position and update, which makes the proposed scheme find routes efficiently based on the network dynamics quickly; therefore, K-MORP offers less delay results compared to other schemes. From Fig. 8, it can be observed that K-MORP offers 46.21% less delay compared to K-means, 56.6% with respect to CPSO, 60% with AODV, 59.6% with OLSR, and 58.2% with PSO. The other two schemes, IHCR has 20% and 33.98% higher delays, respectively, in comparison with K-MORP. Furthermore, K-MORP adopted an enhanced Gauss Markov 3D mobility model, which predicts the position and direction of each UAV in an appropriate manner, thereby improving the network performance and maintaining less delay even on a high number of UAVs.

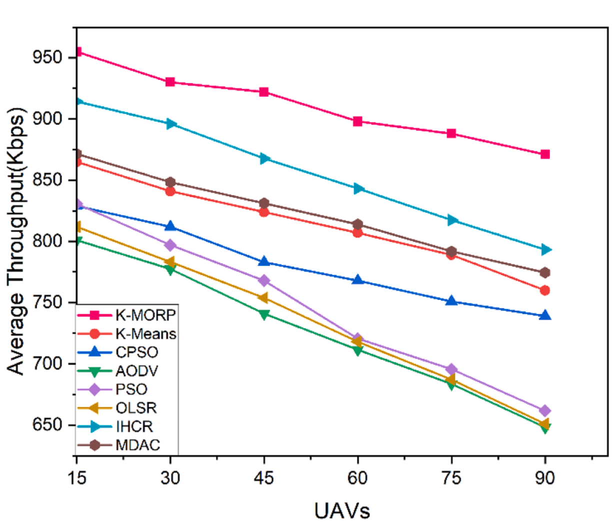

Fig. 9. . Throughput (Kbps) versus the number of UAVs.

D. throughput

Maintaining a high level of throughput in a UAV communication network is crucial for ensuring efficient data transmission. Throughput measures the amount of data(bits) successfully transmitted per unit time from the source UAV node to the destination UAV. Fig. 9 represents the overall average throughput across the network rather than the throughput per node. The average throughput was computed by summing the total bits received across all nodes and dividing by the entire simulation duration. This metric was chosen to provide a broader view of the overall network performance under different scenarios. The proposed scheme, K-MORP, is designed to optimize throughput in a multi-UAV network. The figure shows that the initial throughput value of the K-MORP scheme is 948kpbs. However, as the number of UAVs increases, the throughput decreases. Adding more UAVs creates more traffic and can lead to congestion in the network. Despite this decrease, the K-MORP scheme maintains its efficiency and achieves a throughput value of up to 875kpbs, even at the highest number of UAVs. In contrast, other routing protocols such as AODV, SPO, SCPO, MDAC, and OLSR experience a significant reduction in their throughput value as the number of UAVs increases. In comparison to these schemes, K-MORP offers 10.4% with K-Means, 15.211% with CPSO, 19.204% with AODV, 13.4% with CPSO, and 17.3% with OLSR respectively. This is due to the unstable communication links and dynamic nature of UAVs, which make it challenging for packets to reach their intended destination. Although the IHCR throughput is 914 Kbps at the starting point, its value also decreases as the number of UAVs increases due to the presence of a complicated Gauss Markov 3d mobility model that considers the mean velocity, patch angle, and direction components. However, these schemes are mostly designed for 2D environments and adopt simple random waypoint mobility models, which are not appropriate for UAVs in 3D environments. The figure shows that the throughput of these protocols is approximately 650 Kbps at 90 UAVs, which is significantly lower than the K-MORP scheme's throughput value. The K-MORP scheme achieves high throughput by optimizing the routing paths between the UAV nodes. It uses a hierarchical approach to divide the network into clusters, with each cluster having its cluster head. The cluster heads are responsible for communicating with other cluster heads to exchange information about the network topology and select the best path for data transmission. Using this approach, the K-MORP scheme can minimize the number of hops required for data transmission, reducing the chances of packet loss and improving the overall throughput. The proposed K-MORP scheme maintains a high level of throughput in a UAV communication network, even in the presence of multiple UAVs. Its hierarchical approach to routing allows for efficient data transmission, whereas other protocols struggle to maintain throughput as the number of UAVs increases. As such, the K-MORP scheme has the potential to improve the efficiency and reliability of UAV communication networks significantly.

E. normalized routing load

Fig. 10 shows that the normalized routing load ratio (NRL%) for

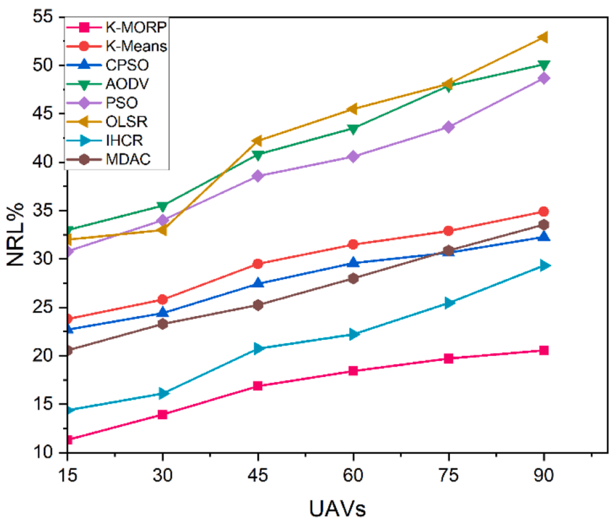

Fig. 10. NRL% versus number of VAVs.

different routing schemes increases with an increasing number of UAVs in the network. As the number of UAVs increases from 0 to 90, the normalized routing load increases for all the routing schemes. This is expected, as more UAVs would mean more control packets for routing, resulting in a higher routing overhead. Among the different routing schemes, the reactive routing protocols AODV and PSO show higher routing loads. Initially, the normalized routing load is 33 for AODV and 30.8% for PSO. As the number of UAVs increases, the NRL% value reaches up to 52.5% for AODV, 50% for PSO and 53.9% for OLSR, respectively. On the other hand, clustering-based routing protocols K-means, MDAC, and IHCR have lower routing load for the same number of UAVs, with 34.9%, 33.55%and 29.33%, respectively. The NRL% value of the proposed K-MORP scheme ranges from 11.33% to 20.56, outperforming all other clustering-based routing schemes. The proposed scheme adopted transition probabilities from a Gaussian distribution to predict the positions and paths of UAVs. Moreover, k-means constraint-based clustering efficiently forms clusters; therefore, the NRL% value is stable. On the other hand, due to the high mobility of UAVs, traditional routing schemes suffer from the formation of optimal clusters, and as a result, some UAVs reside in different clusters; therefore, higher routing packets are generated. Hence, the proposed scheme offers less NRL value as compared to all schemes from Fig. 8. It can be clearly observed that the two benchmark schemes offer 65.6% (AODV) and 64.4% higher NRL values. At the same time, PSO and CPSO are particle swarm-based and offer 63.17% (PSO) and 52.2% (CPSO) higher network loads. The traditional K-means clustering scheme suffers because inefficient clustering has a 52.21% higher routing load. In contrast, MDAC, which is an integration of OLSR and K-means, has a 45.01% higher NRL value than K-MORP. Finally, IHCR has a 21.1% higher NRL value than K-MORP, which shows that the proposed scheme has a lower NRL value and better performance.

4.3. Comparative analysis

Table 4 delineates the performance of each routing protocol across diverse comparative metrics, showing the efficiency inherent in each routing scheme. This table presents a comparative performance analysis of the routing protocols across various parameters, including throughput, packet delivery ratio (PDR), end-to-end delay (E2E), normalized routing load (NRL), and mean square error (MSE). From the

Table 4
Comparative performance table.

Routing Protocols	Compare Metric				
	Throughput	PDR	E2E	NRL	MSE
K-MORP	High	High	Low	Low	Low
K-Means	Medium	High	Medium	Medium	Low
IHCR	High	High	Low	Medium	
MDAC	High	Medium	Medium	Medium	
CPSO	Medium	Medium	Medium	Medium	High
AODV	Low	Low	High	High	
OLSR	Low	Low	High	High	
PSO	Low	Low	High	High	High

table, it can be observed that K-means and MDAC offer the same results because both schemes operate the k-means clustering algorithm, although MDAC has a high throughput. After all, the auto-encoder slightly improves its results. Although IHCR offers the second-best results, as shown in the table. 4. It offers better results in terms of high throughput, high PDR, and low end-to-end delay. However, it has medium normalization because this scheme considers 2D communication in its original form. When implemented in 3D communication, it suffers from a stable path; therefore, an extra overhead is generated for routing. On the other hand, CPSO shows a slight improvement, offering medium results throughout(kbps), PDR%, and E2E delay(s) though having a high mean square error, which shows that CPSO has a slight improvement as compared to PSO. Moreover, topology-based AODV and OLSR have the lowest results because these schemes maintain stable paths due to the high mobility of UAVs and the 3D coverage area. Finally, the proposed scheme possesses the best results because it monitors the UAVs movement efficiently and forms optimal clusters, which avoids overlapping and predicts the position of each UAV accurately in UAV ad-hoc networks.

5. Conclusion

In conclusion, unmanned aerial vehicles (UAVs) have emerged as a promising technology for wireless communication applications due to their flexibility and ability to operate in three-dimensional (3D) space. However, the fast movement of UAVs and the 3D environment presents unique challenges for communication and packet routing in UAV ad hoc networks. To address these challenges, clustering-based UAV ad hoc routing schemes have been proposed that divide the network into clusters to reduce overhead and improve efficiency. However, traditional clustering schemes suffer from issues such as cluster overlapping and real-time monitoring of UAVs. This paper introduces a k-means online learning routing protocol that monitors the real-time movement of UAVs and forms clusters based on real-time data. The Gauss Markov 3D mobility model assists in determining UAV positions and velocity within 3D space. The introduced approach improves the performance and reliability of UAV ad hoc routing protocols by reducing the network load and guiding UAVs to select optimal routes for data transmission. Additionally, the approach is efficient for UAV ad hoc networks, where UAVs are constantly moving and the network topology is changing. Based on the contributions of the proposed approach, it holds the potential to enhance the performance of UAV communication networks significantly, maximizing the full potential of UAV technology. In future work, the proposed scheme can be further modified by adopting advanced machine learning techniques to improve the cluster formation and routing decision-making process. Additionally, energy-efficient strategies can also be designed, such as power allocation, and node sleep/wake scheduling time would be crucial for addressing the energy constraint issue in UAV ad hoc networks (Fig. 1).

Data availability

The data will be available upon request to the corresponding author.

Code availability

The code will be available upon request to the corresponding author.

Declaration of Competing Interest

The authors declare no conflict of interest.

Data availability

Data will be made available on request.

Acknowledgement

The authors are highly grateful to their affiliated universities and institutes for providing research facilities and funding for this research work.

References

[1] H. Wang, H. Zhao, J. Zhang, D. Ma, J. Li, J. Wei, Survey on unmanned aerial vehicle networks: a cyber physical system perspective, IEEE Commun. Surv. Tutorials 22 (2) (2020) 1027–1070.
[2] R. Nagasawa, E. Mas, L. Moya, et al., Model-based analysis of multi-UAV path planning for surveying postdisaster building damage, Sci. Rep. 11 (2021) 18588.
[3] Arnau Rovira-Sugranes, Abolfazl Razi, Fatemeh Afghah, Jacob Chakareski, A review of AI-enabled routing protocols for UAV networks: trends, challenges, and future outlook, Ad Hoc Netw. 130 (2022), 102790. ISSN 1570-8705.
[4] Q. Sang, H. Wu, L. Xing, H. Ma, P. Xie, An energy-efficient opportunistic routing protocol based on trajectory prediction for FANETs, IEEE Access 8 (2020) 192009–192020, https://doi.org/10.1109/ACCESS.2020.3032956.
[5] Amira Chriki, Haifa Touati, Hichem Snoussi, Farouk Kamoun, FANET: communication, mobility models and security issues, Comput. Networks 163 (2019), 106877.
[6] L. Zhang, F. Hu, Z. Chu, E. Bentley, and S. Kumar, "3D transformative routing for UAV swarming networks: a skeleton-guided, GPS free approach," IEEE Trans. Veh. Technol., vol. 70,no. 4, pp. 3685.
[7] N. Zhao, Z. Ye, Y. Pei, Y.-C. Liang, D. Niyato, Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing, IEEE Trans. Wireless Commun. 2022 (2021) 3701.
[8] B. Li, S. Zhao, R. Miao, R. Zhang, A survey on unmanned aerial vehicle relaying networks, IET Commun. 15 (10) (2021) 1262–1272.
[9] F. Zhou, R.Q. Hu, Z. Li, Y. Wang, Mobile Edge Computing in Unmanned Aerial Vehicle Networks, IEEE Wirel. Commun. 27 (1) (February 2020) 140–146, https://doi.org/10.1109/MWC.001.1800594.
[10] J. Jiang, G. Han, Routing protocols for unmanned aerial vehicles, IEEE Commun. Mag. 56 (2018) 58–63.
[11] H. Touati, A. Chriki, H. Snoussi, F. Kamoun, Cognitive Radio and Dynamic TDMA for efficient UAVs swarm communications, Comput. Netw. 196 (2021), 108264.
[12] C. Jiang, Z. Hu, Z.P. Mourelatos, D. Gorsich, P. Jayakumar, Y. Fu, M. Majcher, R2-RRT*: reliability-based robust mission planning of off-road autonomous ground vehicle under uncertain terrain environment, IEEE Trans. Autom. Sci. Eng. 19 (2021) 1030–1046.
[13] L. Hong, H. Guo, J. Liu, Y. Zhang, Toward swarm coordination: topology-aware inter-UAV routing optimization, IEEE Trans. Veh. Technol. 69 (2020) 10177–10187.
[14] R. Shanthy, T. Padma, A zone routing protocol incorporated with sleep scheduling for MANETs, J. Ambient. Intell. Humaniz. Comput. 12 (2021) 4181–4191.
[15] B. Zheng, K. Zhuo, H. Zhang, H.-X. Wu, A novel airborne greedy geographic routing protocol for flying ad hoc networks, Wireless Networks (2022) 1–15.
[16] Y. Cui, Q. Zhang, Z. Feng, Z. Wei, C. Shi, H. Yang, Topologyaware resilient routing protocol for fanets: an adaptive Q-learning approach, IEEE Int. Things Journal 9 (19) (2022) 18.632–18.649.
[17] X. Wang, Y. Lu, Efficient forwarding and data acquisition in NDN-based MANET, IEEE Trans. Mob. Comput. 21 (2) (2022) 530–539.
[18] M.Y. Arafat, S. Moh, A Q-learning-based topology-aware routing protocol for flying ad hoc networks, IEEE Int. Things J. 9 (3) (2021) 1985–2000.
[19] Q. Wu, et al., Routing protocol for heterogeneous FANETs with mobility prediction, China Commun. 19 (1) (Jan. 2022) 186–201, https://doi.org/10.23919/JCC.2022.01.014.
[20] Konstantinos Kanistras, Goncalo Martins, Matthew J. Rutherford, Kimon P. Valavanis, A survey of unmanned aerial vehicles (UAVs) for traffic monitoring, in: 2013 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, 2013, pp. 221–234.
[21] Jacob. Chakareski, UAV-IoT for next generation virtual reality, IEEE Trans. Image Process. 28 (12) (2019) 5977–5990.
[22] Awadhesh Dixit, Sunil Kumar Singh, BMUDF: hybrid bio-inspired model for fault-aware UAV routing using destination-aware fan shaped clustering, Int. Things 22 (2023), 100790.
[23] Xiaohan Qiu, Shan Zhang, Zhiyuan Wang, Hongbin Luo, Integrated Host-and Content-Centric Routing for Efficient and Scalable Networking of UAV Swarm, IEEE Trans. Mob. Comput. (2023).
[24] Feng Wang, Heng Zhang, Aimin Zhou, A particle swarm optimization algorithm for mixed-variable optimization problems, Swarm Evol Comput 60 (2021), 100808.
[25] Ali Abbas, Bhawani Shankar Chowdhry, Muhammad Saqib, Vishal Dattana, Dynamic routing and coordination of cluster for unmanned aerial vehicle (UAV) swarms, Math. Probl. Eng. 2021 (2021) 1–11.
[26] Y.Volkan Pehlivanoglu, Perihan Pehlivanoglu, An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems, Appl. Soft Comput. 112 (2021), 107796.
[27] Jianmin Liu, Qi Wang, ChenTao He, Katia Jaffrès-Runser, Yida Xu, Zhenyu Li, YongJun Xu, QMR:q-learning based Multi-objective optimization Routing protocol for Flying Ad Hoc Networks, Comput. Commun. 150 (2020) 304–316. ISSN 0140-3664.
[28] S. Ullah, K.H. Mohammadani, M.A. Khan, Z. Ren, R. Alkanhel, A. Muthanna, U. Tariq, Position-monitoring-based hybrid routing protocol for 3D UAV-based networks, Drones 6 (11) (2022) 327.
[29] Z.Ren Saifullah, K.H. Mohammadani, W. Riaz, Optimal Game Routing for UAV Adhoc Networks in Smart City, in: 2023 6th World Conference on Computing and Communication Technologies (WCCCT), Chengdu, China, 2023, pp. 34–38.
[30] J. Guo, et al., ICRA: an Intelligent Clustering Routing Approach for UAV Ad Hoc Networks, IEEE Trans. Intell. Transp. Syst. 24 (2) (Feb. 2023) 2447–2460.
[31] J. Lansky, A.M. Rahmani, M.H. Malik, et al., An energy-aware routing method using firefly algorithm for flying ad hoc networks, Sci. Rep. 13 (2023) 1323.
[32] Xiaopeng Tan, Zhen Zuo, Shaojing Su, Xiaojun Guo, Xiaoyong Sun, Research of security routing protocol for UAV communication network based on AODV, Electronics (Basel) 9 (8) (2020) 1185.
[33] M.F. Khan, I.N.D.R.A.N.I. Das, Critical analysis of modified gauss markov mobility model using varying values of parameters to check the impact of QoS In MANET, J. Eng. Sci. Technol. 17 (5) (2022) 3393–3409.
[34] A. Raza, M.F. Khan, M. Maqsood, B. Haider, F. Aadil, Adaptive k-means clustering for flying ad-hoc networks, KSII Trans. Int. Inf. Syst. 14 (6) (2020) 2670–2685.
[35] X. Tan, Z. Zuo, S. Su, X. Guo, X. Sun, D. Jiang, Performance analysis of routing protocols for UAV communication networks, IEEE Access 8 (2020) 92212–92224.
[36] J. Fang, C. Wang, R. Li, H. Wei, M. Zhao, Matrix factorization and deep autoencoder based clustering scheme for large-scale UAV networks, in: 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), Florence, Italy, 2023, pp. 1–6, https://doi.org/10.1109/VTC2023-Spring57618.2023.10200587.
[37] Available online: https://www.nsnam.org/releases/ns-3-35/(accessed on 1 March 2023).

Mr. SAIFULLAH received his B.E (Bachelor of Engineering) degree from Mehran University Engineering, Jamshoro-Pakistan. He is currently pursuing a Ph.D. degree in the School of Information and Communication Engineering, at Chongqing University of Post and Telecommunication, Chongqing-China. His research interests are UAV networks, routing protocols, game theory algorithms, and future wireless networks. Email: l202010009@stu.cqupt.edu.cn

Prof. Zhi Ren received the B.S. degree in applied electronics from Southwest Jiaotong University, Chengdu, China in 1993, and the M.S. and Ph.D. degrees in measuring and testing technology and communication and information systems from the University of Electronic Science and Technology of China in 2002 and 2005, respectively. From 2006 to 2008, he was a postdoctoral research associate in the Department of Electrical and Computer Engineering, Stevens Institute of Technology, NJ, USA. He is now a professor in the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications of China. His research interests include wireless mobile networks and terahertz communications. Email: renzhi@cqupt.edu.cn

Dr. Khalid H. Mohammadani (Member, IEEE) received the B.S. degree in telecommunication from the University of Sindh, Jamshoro, Sindh, Pakistan, in 2008, the M.Phil. degree in computer science from Isra University, Hyderabad, Sindh, in 2015, and the Ph.D. degree from the State Key Laboratory of Information Photonics and Optical Communications (IPOC), School of Electronic Engineering, Beijing University of Posts and Telecommunications, China, in 2022. He served one year as an Adjunct Lecturer with the Department of Computer Science, Huanggang Normal University, Hubei, China, 2022–2023. He has more than five years experience in teaching for Telecommunication and Computer Science subjects in the University of Sindh and ISRA university Hyderabad, Pakistan. He is currently a Post Doc with the Electro-Optical Communication(ECO) group, Department of Electrical Engineering, Eindhhoven University of Technology (TU/e). His research interests include several areas related to optical and wireless networks, such as passive optical network (PON), PON virtualization for 5 G and beyond, optical metro access networks, fiber wireless networks, dynamic bandwidth and wavelength allocation (DBA/DWA), AI for optical networks, network security, hybrid wireless networks, routing protocols of 2D (MANET/VANET) and 3D unmanned aerial vehicle (UAV) wireless networks, and the Internet of Things (IoT). He has published several papers on these topics in IEEE, Elsevier, and Springer journals and presented his research at several IEEE conferences. He is involved in both the Computer Science and Photonics Societies, Institute of Electrical and Electronics Engineers (IEEE). Email: k.hussain@tue.nl

Prof. M Faheem received the B.Sc. Computer Engineering degree in 2010 from the Department of Computer Engineering at the University College of Engineering & Technology, Bahauddin Zakariya University Multan, Pakistan. In 2012, he received an MS in Computer Science from the Faculty of Computer Science and Information Systems at Universiti Teknologi Malaysia. He received his Ph.D. in Computer Science from the Faculty of Engineering, Universiti Teknologi Malaysia, Johor Bahru, Malaysia in 2021. Previously, he served as a lecturer at Comsats Institute of Information & Technology from 2012 to 2014, in Pakistan. Also, he worked as an Assistant Professor in the Department of Computer Engineering at Abdullah Gul University from 2014 to 2022, in Turkey. Currently, he is a Sr. researcher in the School of Computing (Innovations and Technology) at the University of Vaasa, Vaasa, Finland. His-research interest includes the areas of cybersecurity, blockchain, artificial intelligence, smart grid, and smart cities. Dr. Faheem has authored several papers in refereed journals and conferences, and served as a reviewer for numerous journals in IEEE, Elsevier, Springer, Willey, Hindawi, and MDPI. Email: muhammad.faheem@uwasa.fi