Journal of Theoretical and Applied Information Technology
15th July 2025. Vol.103. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DRAGONFLY-INSPIRED OLSR PROTOCOL FOR INTERFERENCE-AWARE SEAMLESS CONNECTIVITY IN FANET

GUNAVATHY H[1], Dr. RAVINDRANATH P.V[2]
[1]Research Scholar, School of Computer Science (PG), RVS College of Arts & Science, Sulur, Coimbatore, Tamil Nadu, India
[2]Assistant Professor, School of Computer Science (PG), RVS College of Arts & Science, Sulur, Coimbatore, Tamil Nadu, India
.E-mail: [1]gunahg.2014"gmail.com ,[2]ravindranath@rvsgroup.com

ABSTRACT

Flying Ad Hoc Networks (FANETs) are vital for various applications, relying on unmanned aerial vehicles (UAVs) for communication in dynamic and challenging environments. Seamless connectivity within FANETs is crucial for uninterrupted data exchange and mission success. However, interference from terrain obstacles, weather conditions, and other UAVs poses significant challenges to routing efficiency. The proposed "Dragonfly-Inspired OLSR Protocol (DO-OLSR)" introduces a novel approach by integrating dragonfly-inspired optimization techniques into the OLSR protocol. This integration minimizes control overhead and route discovery latency, optimizing network performance. The protocol incorporates interference-aware mechanisms that dynamically adapt routing decisions based on real-time environmental conditions, mitigating interference effects. Through simulation-based evaluations, the protocol demonstrates improved network performance, reduced packet loss, and enhanced throughput compared to traditional routing protocols. By dynamically adapting to real-time environmental conditions, the DO-OLSR maintains seamless connectivity while mitigating interference effects, showcasing its potential to enhance overall network reliability and performance in FANETs.

Keywords: *FANET, UAV, Seamless Connectivity, Interference Mitigation, Dragonfly Optimization, OLSR Protocol*

1. INTRODUCTION

Flying ad hoc networks (FANETs) are a specialized form of mobile ad hoc networks (MANETs) designed to facilitate communication and coordination among unmanned aerial vehicles (UAVs) in dynamic aerial environments. Unlike traditional terrestrial networks, FANETs operate without needing fixed infrastructure, allowing UAVs to communicate and collaborate autonomously[1]. This decentralized approach enables FANETs to be deployed rapidly in various scenarios, including surveillance, reconnaissance, environmental monitoring, and disaster response. By leveraging the mobility and agility of UAVs, FANETs offer unique capabilities such as aerial surveillance, remote sensing, and real-time data collection over large and often inaccessible areas[2], [3]. Integrating FANETs with the Sophisticated Rapid Response System (SRRS) enhances their effectiveness in scenarios requiring proactive threat detection and rapid response coordination. Overall, FANETs represent a versatile and adaptable communication platform for UAVs, with applications across various domains[4].

Seamless connectivity is paramount in FANETs and UAVs, serving as the linchpin for uninterrupted data transmission and enabling seamless interaction between UAVs and ground stations. Beyond mere convenience, it is essential for enhancing operational efficiency and effectiveness across myriad applications, from aerial surveillance and reconnaissance to disaster response and environmental monitoring[5]. Moreover, seamless connectivity is pivotal for enabling emerging UAV technologies like autonomous navigation, collaborative sensing, and swarm coordination, which rely on constant and reliable communication for effective operation. From precision agriculture to infrastructure inspection, seamless connectivity underpins mission-critical UAV applications, enabling real-time data collection, situational awareness, and decision-making[6]. As the demand for UAV-based solutions continues to soar, ensuring seamless connectivity remains a top priority, driving innovation and unlocking the full potential of FANETs and UAVs in various domains[7]. The difference between UAV and FANET is provided in Table 1.

Routing is a linchpin within the Sophisticated SRRS, pivotal in orchestrating an agile and efficient response to security challenges. As the backbone of communication and resource allocation, routing algorithms within the SRRS ensure that information flows seamlessly between various components, optimizing the deployment of unmanned vehicles, sensors, and response teams[8]. This dynamic routing capability allows the SRRS to adapt swiftly to evolving threats, directing resources to critical areas with precision[9]. Furthermore, the efficiency of routing mechanisms directly impacts the system's ability to harness real-time data, enabling rapid decision-making and coordination. The significance of routing within the SRRS lies in its capacity to transform raw data into actionable intelligence, ensuring a proactive and well-coordinated response to safeguard borders and respond effectively to emergencies[10], [11].

Table 1. Difference Between UAV and FANET

Aspect	UAV	FANET
Definition	Single unmanned aircraft capable of autonomous flight without an onboard pilot.	A network of interconnected UAVs that communicate with each other to achieve common goals.
Purpose	Various applications such as surveillance, reconnaissance, aerial photography, package delivery, agriculture, search and rescue, and environmental monitoring.	Enable communication and coordination between multiple UAVs to form a network.
Communication	Point-to-point with ground station.	Multi-hop between UAVs in the network.
Coordination	Limited but operates in an independent manner	Decentralization enabled collaboration.
Scalability	Limited to individual UAV capabilities.	Scalable with the addition of more UAVs.
Redundancy	Relies on backup systems for failures.	Inherent redundancy with multiple UAVs.
Collaboration	Minimal, each UAV operates solo.	Facilitates cooperative task-sharing.
Application Diversity	Various industries, diverse tasks.	Suited for collective tasks like SAR.
Network Infrastructure	Relies on existing or dedicated ground stations.	Requires specialized multi-hop protocols.

The study presumes that all UAVs are equipped with uniform communication hardware and operational capabilities, promoting consistent behavior across the network. It assumes that environmental conditions simulated within the study mirror a range of realistic scenarios, although they simplify the inherent unpredictability of actual environments. Moreover, it is presumed that all UAVs operate in compliance with existing regulatory frameworks without interference from non-cooperative entities.

The research primarily focuses on simulated environments, which may not encompass all the complexities of real-world operations. Energy consumption metrics are based on theoretical calculations, which may not accurately represent the actual energy use due to unforeseen operational inefficiencies. Additionally, the protocol's adaptability to scenarios involving extremely high UAV densities or significant heterogeneity in UAV capabilities remains untested, which could affect its applicability in diverse operational contexts.

1.1. Problem Statement

FANETs have become integral to a wide range of applications, including environmental monitoring, disaster response, and strategic surveillance. Despite significant advancements, these networks continue to grapple with challenges such as high mobility dynamics, interference from complex terrains, and stringent energy constraints. These factors critically impact the operational longevity and efficiency of unmanned aerial vehicles (UAVs). Existing routing protocols often fail to simultaneously address the demands for dynamic adaptability, energy efficiency, and minimized control overhead in environments characterized by high mobility and significant interference. This deficiency highlights the urgent need for a routing protocol that enhances communication reliability, improves network throughput, and extends the operational capabilities of UAVs by reducing energy consumption and adapting in real-time to changing environmental conditions.

1.2. Motivation

The motivation for tackling the energy consumption issue in interference-aware routing for FANETs lies in the critical role that UAVs play in various applications, including surveillance, disaster response, and environmental monitoring. Optimizing energy consumption through efficient routing protocols enhances UAV operational endurance, enabling longer-duration missions and broader application coverage. Reducing energy consumption can prolong UAV operational lifespan, improve mission reliability, and enhance responsiveness to dynamic airspace conditions. Moreover, efficient energy utilization contributes to environmental sustainability by minimizing the need for frequent recharging or battery replacements. Addressing this challenge not only advances the capabilities and reliability of FANETs but also facilitates the widespread adoption and deployment of UAV technologies, ultimately benefiting society through improved safety, efficiency, and effectiveness in critical operations.

1.3. Objective

To develop and evaluate an energy-efficient Dragonfly-inspired OLSR (Optimized Link State Routing) protocol tailored for interference-aware seamless connectivity in FANETs. This objective addresses the challenge of managing energy consumption in UAV operations within dynamic airspace environments while optimizing routing decisions to mitigate interference effects and enhance network performance. The research will focus on designing innovative routing algorithms that dynamically adapt to changing network conditions, optimize routing paths based on energy consumption metrics, and incorporate mechanisms for energy-efficient communication and resource management. Through this objective, the research aims to extend UAV operational lifespan, improve mission endurance, and facilitate the widespread adoption of UAV technologies for various critical applications, including surveillance, disaster response, and environmental monitoring.

2. LITERATURE REVIEW

"Dynamic AeroNet Routing"[12] adapts the virtual network structure based on real-time conditions, making optimal communication in dynamic airborne environments. It dynamically adjusts the connectivity between nodes, responding to changes in network parameters. It minimizes latency and maximizes resource utilization by enabling routing efficiency. The network's resilience is focused on continuous optimization of the virtual topology. "ClusterFly Protocol"[13] operates on a cluster-based routing mechanism. Nodes dynamically form clusters, typically led by a designated cluster head. The clustering process considers factors like proximity and communication capabilities. Each cluster head manages intra-cluster communication, reducing the need for direct communication between all nodes. Cluster heads handle inter-cluster communication, enhancing scalability and minimizing the impact of high mobility in FANETs. "3D GeoRoute Protocol"[1] dynamically determines their effective transmission ranges, accounting for the complexities of aerial communication. It progresses the routing by establishing optimized routes considering horizontal and vertical dimensions. Nodes strategically choose neighbours within their adjusted transmission range for forwarding, enhancing overall routing efficiency.

"ICRA Protocol"[14] presents a sophisticated method for enhancing communication efficiency. The protocol employs intelligent clustering, where UAVs autonomously form clusters based on factors such as proximity and communication parameters. Within these clusters, intelligent routing mechanisms optimize communication paths. "Dyna-Posi-OLSR"[2] revolves around continuously exchanging real-time positional data among nodes. Nodes broadcast their current positions regularly, contributing to constructing a dynamic topology map. The protocol

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

utilizes a modified OLSR algorithm that leverages this spatial awareness to make informed decisions for optimizing routing paths. "ElasticFly Routing"[15] involves nodes continuously monitoring real-time conditions such as airspace congestion and node mobility. Each node adjusts its routing path elastically, optimizing for factors like available bandwidth and minimizing latency. It ensures the communication infrastructure in FANET remains responsive and efficient, overcoming the unpredictable nature of aerial environments.

"QoS-RL VehiRoute"[16] utilizes intersections as decision points for route optimization. Through continuous reinforcement learning, nodes evaluate different routes based on QoS metrics such as latency and reliability. The protocol refines its decision-making process, learning to adapt routing paths dynamically. "FT-AdHoc On-Demand"[17] employs an on-demand routing strategy with built-in fault tolerance mechanisms. Nodes initiate route discovery only when needed, conserving resources. In the event of a link or node failure, the protocol dynamically triggers route recalculations, ensuring continuous communication. By integrating fault tolerance directly into the on-demand routing process, the protocol enhances the resilience of Mobile Ad Hoc Networks (MANET). "FD-VANET End-to-End Delay"[18] meticulously minimizes communication delays through simultaneous transmission and reception capabilities. The nodes, equipped with full-duplex radios, engage in dynamic optimization of communication paths by actively assessing real-time traffic conditions and mitigating signal interference. This intricate working mechanism ensures an agile adaptation of routes, continually optimizing to minimize end-to-end delays.

"UAV-OptiRoute Systematic Review"[19] employs a systematic review approach to discern the technical efficiency of various bio-inspired algorithms. It intricately involves the application of algorithms inspired by nature, such as swarm intelligence or genetic algorithms, for the dynamic adaptation of UAV routes contingent on environmental conditions. "Multi-Obj Packet Routing"[20] is tailored for aeronautical ad-hoc networks (AANETs). It orchestrates packet routing efficiency through a sophisticated working mechanism in real-time multi-objective optimization. It dynamically balances and optimizes conflicting objectives, minimizing latency, maximizing throughput, and ensuring energy efficiency. It showcases technical prowess by adapting routing decisions to the unique challenges posed by AANETs, where high mobility and variable link conditions demand constant and dynamic adjustments. "DRL-QoS CR-MANETs"[21] integrate Deep Reinforcement Learning, empowering nodes to autonomously learn optimal routing decisions based on real-time interactions with the environment. Incorporating cross-layer design enriches the decision-making process by assimilating information from multiple protocol layers, resulting in a technically advanced and adaptive approach to changing network conditions.

"Prediction-supported Adaptive Routing (PAR)"[22]is a sophisticated integration of prediction-supported adaptive routing and deep reinforcement learning (DRL) tailored for FANETs. Initially, the protocol employs predictive models that leverage historical data and network parameters to anticipate potential fluctuations in network conditions. These predictive models equip drones with the foresight to adjust their routing paths preemptively. Subsequently, drones utilize DRL algorithms to iteratively refine their routing decisions based on real-time feedback obtained from the environment. Drones optimize their routing strategies through continuous reinforcement learning, ensuring efficient data transmission amidst the dynamic aerial environment. PARouting's adaptive and self-learning mechanism guarantees robust and reliable communication within FANETs, paving the way for enhanced performance and adaptability in various applications.

"Eagle Optimized Energy Efficient Optimal Route-Finding Protocol (EOEEORFP)"[23]is proposed to enable data transmission within the dynamic environment of FANETs. Initially, the protocol assesses the network topology and the energy levels of participating drones to determine the most efficient routing paths. Utilizing optimization algorithms, EOEEORFP evaluates multiple potential routes based on distance and energy consumption criteria. Once the optimal routes are identified, EOEEORFP implements robust security measures to protect data during transmission. These security mechanisms may include encryption algorithms, authentication protocols, and intrusion detection systems to mitigate potential threats. By seamlessly integrating optimization algorithms with stringent security protocols, EOEEORFP ensures reliable and secure data transmission in FANETs, enhancing aerial communication networks' overall functionality and resilience. Many bio-inspired optimization algorithms are applied in ad-hoc networks for achieving the better results [24]-[62].

2.1. Critique

Existing FANET protocols often excel in controlled simulations but struggle under the dynamic conditions typical of real-world UAV operations, raising concerns about their scalability and efficiency. Many claims of enhanced energy efficiency are based on idealized assumptions that rarely hold in operational scenarios, pointing to a gap between simulated results and actual performance. Furthermore, the reliance on outdated methods and the lack of innovative approaches in the literature highlight a significant disconnect between theoretical improvements and practical usability. This critique emphasizes the necessity for protocols like the DO-OLSR, designed to address these shortcomings by providing robust performance and real-time adaptability in the complex and unpredictable environments where FANETs operate.

3. DRAGONFLY OPTIMIZATION-INSPIRED OLSR PROTOCOL (DO-OLSR)

The Dragonfly Optimization-Inspired OLSR Protocol merges nature's efficiency with modern networking. Drawing inspiration from the swift and adaptive flight of dragonflies, this protocol optimizes the performance of the OLSR algorithm. Like dragonflies efficiently navigate complex environments, this protocol enhances routing in dynamic ad hoc networks. It maximizes throughput and minimizes latency by dynamically adjusting routes based on network changes and optimizing message exchange. Its decentralized nature mirrors the autonomy of dragonflies, ensuring robustness even in challenging network conditions. The fusion of biological insights and technological innovation heralds a new era in efficient communication protocols.

Research Hypothesis:

Based on the identified gaps and the objectives outlined in the study, the following hypotheses are articulated to guide the experimental validation and theoretical assertions of the Dragonfly-Inspired OLSR Protocol:

- Hypothesis 1 (H1): The integration of dragonfly-inspired optimization algorithms within the OLSR protocol significantly reduces control overhead and routing latency compared to traditional OLSR and other commonly used routing protocols in FANETs.
- Hypothesis 2 (H2): DO-OLSR enhances network performance in terms of throughput, packet delivery ratio, and latency under dynamic environmental conditions, performing superiorly compared to existing routing protocols employed in similar FANET settings.
- Hypothesis 3 (H3): The implementation of the DO-OLSR protocol results in a noticeable improvement in energy efficiency, thereby extending the operational endurance of UAVs within FANETs without compromising network reliability and communication quality.
- Hypothesis 4 (H4): DO-OLSR exhibits greater adaptability to sudden changes in network topology and environmental conditions, demonstrating more robust and reliable communication in FANETs compared to conventional routing protocols.

These hypotheses will be tested through a series of simulations and empirical evaluations, designed to rigorously assess the performance, efficiency, and adaptability of the DO-OLSR protocol under varied and controlled experimental conditions. The outcomes of these tests are anticipated to validate the proposed benefits of the protocol, substantiating its potential to revolutionize communication strategies within FANETs.

3.1. Objective Definition

Objective definition is a crucial initial step in DO-OLSR. In this step, the optimization problem is precisely formulated, specifying the algorithm's goal. Mathematically, the objective function is denoted as $f(x)$, where x represents the solution vector, i.e., the configuration of the OLSR routing protocol. In the context of OLSR, the objective function encapsulates the performance metrics that the optimization seeks to optimize. Let P be the set of all possible OLSR configurations and x_i be the configuration of the i^{th} dragonfly in the population. The objective function $f(x_i)$ quantifies the quality of the OLSR configuration and is specific to the optimization goal. For instance, if the goal is to minimize end-to-end delay, the objective function may be expressed as Eq.(1).

$$f(x_i) = \sum_{j=1}^{N} D_j(x_i) \tag{1}$$

where $D_j(x_i)$ represents the end-to-end delay for the j^{th} link in the network, and N is the total number of links. The objective function aggregates the delays over all links to produce a scalar value representing the overall performance of the OLSR configuration.

Alternatively, if the goal is maximizing network throughput, the objective function could be Eq.(2).

$$f(x_i) = \sum_{j=1}^{N} T_j(x_i) \tag{2}$$

where $T_j(x_i)$ denotes the throughput of the j^{th} link. The negative sign is introduced since Dragonfly Optimization seeks to minimize the objective function, and maximizing throughput is equivalent to minimizing the negative of throughput.

3.2. Parameter Initialization

Parameter initialization is a critical phase in the application of DO-OLSR. It involves setting up the necessary parameters that govern the behaviour of the optimization algorithm. Mathematically, let P be the parameter space, and p represent a vector containing the algorithm-specific parameters. The parameters are crucial in shaping the exploration and exploitation characteristics of the Dragonfly Optimization algorithm. The primary parameters to be initialized include the population size (pop_size), the maximum number of iterations ($max\_iterations$), and any other control parameters specific to Dragonfly Optimization, denoted as, $p = [pop\_size, max\_iterations, \dots]$. The population size determines the number of dragonflies in each generation, influencing the diversity of solutions explored. The maximum number of iterations defines the stopping criterion, ensuring the algorithm terminates after a predefined number of iterations. In mathematical terms, the parameter initialization can be represented as Eq.(3).

$$p^{(0)} = Initialize\_parameters() \tag{3}$$

where $p^{(0)}$denotes the initial parameter vector, and $initialize\_parameters()$ is a function that sets the values for the parameters based on the problem requirements and algorithm characteristics.

Dragonfly Optimization may have specific parameters related to the algorithm's internal dynamics. These could include parameters controlling the influence of leaders on the rest of the population, the step size in the search space, or any adaptation mechanisms. Let q represent the internal parameters of Dragonfly Optimization, and p and q together define the entire set of parameters, and it is shown as Eq.(4).

$$p^{(0)} = [pop\_size, \max\_iterations, q] \tag{4}$$

In enhancing the OLSR protocol, careful parameter tuning is essential to balance exploration and exploitation. The population size influences the varied routing configurations explored, while the maximum number of iterations ensures that the algorithm does not run indefinitely.

3.3. Topology Representation

Topology representation is a pivotal step in DO-OLSR. The FANET's physical layout is mathematically modelled in this step, defining the nodes and links that constitute the communication infrastructure. The representation allows for the translation of the real-world network into a mathematical framework, facilitating the application of optimization algorithms. Consider a network with N nodes denoted as $N = \{1,2,\dots,N\}$, where each node corresponds to a network device capable of transmitting and receiving data. The communication links between nodes form the network's edges, constituting the set of links L. Mathematically, the network topology is represented by a graph $G = (N, L)$, where N is the set of nodes, and L is the set of links. Let A be the adjacency matrix representing the connectivity of the network. The adjacency matrix is a square matrix of size $N \times N$, where A_{ij} is defined as Eq.(5).

$$\begin{cases} 1 & if\ there\ is\ a\ link\ between\ nodes\ i\ and\ j \\ 0 & otherwise \end{cases} \tag{5}$$

Eq.(5) captures the relationships between nodes regarding links, providing a binary representation of network connectivity. The link weight matrix W is defined to represent the characteristics of each link, such as latency or throughput. If w_{ij} represents the weight of the link between nodes i and j, then W is a matrix where $W_{ij} = w_{ij}$ if there is a link, and $W_{ij} = 0$ if there is no link. Eq.(6) determines the link weight between nodes.

$$\begin{bmatrix} w_{11} w_{12} & \dots & w_{1N} \\ w_{21} w_{22} & \dots & w_{2N} \\ \vdots \quad \vdots & \ddots & \vdots \\ w_{N1} w_{N2} & \dots & w_{NN} \end{bmatrix} \quad (6)$$

The OLSR protocol enhancement involves finding the optimal configuration for routing messages among the network nodes. This configuration is represented by a matrix R that defines the routing decisions for each node pair. If r_{ij} is the routing decision for the link between nodes i and j, then R is a matrix where $R_{ij} = r_{ij}$ if there is a link, and $R_{ij} = 0$ if there is no link. Eq.(7) assists in making routing decisions.

$$\begin{bmatrix} r_{11} r_{12} & \dots & r_{1N} \\ r_{21} r_{22} & \dots & r_{2N} \\ \vdots \quad \vdots & \ddots & \vdots \\ r_{N1} r_{N2} & \dots & r_{NN} \end{bmatrix} \quad (7)$$

The objective of Dragonfly Optimization is to optimize the configuration matrix R to improve the overall performance of the OLSR protocol concerning the defined objectives. This involves finding the optimal routing decisions that minimize or maximize the objective function. Considering the dynamic nature of FANET, where link conditions may change over time, the optimization process might include adapting the routing decisions based on the current network state. Let t represent the time index, and $R(t)$ be the configuration matrix at time t. The adaptation of routing decisions over time can be captured by incorporating the time index into the optimization process, expressed in Eq.(8).

$$\begin{bmatrix} r_{11}(t) & r_{12}(t) & \dots & r_{1N}(t) \\ r_{21}(t) & r_{22}(t) & \dots & r_{2N}(t) \\ \vdots & \vdots & \ddots & \vdots \\ r_{N1}(t) & r_{N2}(t) & \dots & r_{NN}(t) \end{bmatrix} \quad (8)$$

3.4. Fitness Evaluation

Fitness evaluation is a crucial step of the DO-OLSR protocol. This step involves assessing the quality of potential solutions represented by dragonflies in the population. The fitness of each solution is determined by how well it aligns with the optimization objectives, providing a quantitative measure of its performance within the context of the OLSR protocol. Let P represent the solution space and x_i denote the i^{th} solution (configuration) within the population. The fitness function, denoted as $f(x_i)$, quantifies the quality of the OLSR configuration encoded by x_i. The objective is to minimize or maximize the fitness function, depending on the optimization goal, expressed as Eq.(9).

$$f(x_i) \quad (9)$$

The specific form of the fitness function depends on the chosen optimization objectives. In the context of OLSR, it may encompass various performance metrics, such as minimizing end-to-end delay, maximizing network throughput, or optimizing for a combination of multiple objectives. Let M represent the set of performance metrics and $M_k(x_i)$ be the k^{th} metric associated with the i^{th} solution. Eq.(10) shows how the fitness function depends on optimization objectives.

$$f(x_i) = \sum_{k \in M} M_k(x_i) \quad (10)$$

Eq.(10) aggregates the contributions of individual metrics, reflecting the holistic evaluation of the OLSR configuration. It transforms the performance metrics into a single scalar value, enabling the algorithm to compare and rank different solutions within the population. If the optimization problem involves constraints, they can be incorporated into the fitness function using penalty terms. Let $C_j(x_i)$ represent the j^{th} constraint associated with the i^{th} solution. The penalized fitness function $\bar{f}(x_i)$ can be expressed as Eq.(11).

$$\bar{f}(x_i) = f(x_i) + \Sigma_j P_j . penalty\left(C_j(x_i)\right) \quad (11)$$

where P_j is a penalty coefficient associated with the j^{th} constraint, and penalty(.) is a function that increases as the constraint violation becomes more severe.

The fitness evaluation process requires computing the fitness values for all dragonflies in the population. Let D represent the set of dragonflies, and F denote the set of fitness values corresponding to each dragonfly, which is expressed in Eq.(12).

$$F = \{f(x_i) | x_i \in D\} \quad (12)$$

The resulting set F provides a basis for the sorting process in subsequent steps, allowing the identification of leaders and guiding the exploration of the solution space. Considering the dynamic nature of FANET, where link conditions may change over time, the fitness evaluation may be performed at each time step t. In this case, the

Eq.(13) becomes a function of both the solution vector x_i and the time index t.

$$f(x_i, t) = \sum_{k\in M} M_k(x_i, t) \tag{13}$$

Incorporating time as a variable enables the algorithm to adapt to changing network conditions and optimize routing configurations dynamically.

3.5. Dragonfly Initialization

Dragonfly initialization involves the creation of an initial population of dragonflies, each representing a potential solution in the search space. The mathematical representation of dragonflies, their positions, and the associated fitness values lay the groundwork for subsequent exploration and optimization. Consider a population of N dragonflies denoted as $D = \{x_1, x_2, \dots, x_N\}$, where x_i represents the i^{th} dragonfly and is a vector encoding the OLSR configuration. The dragonfly population is a diverse set of potential solutions, and each dragonfly's position is crucial in determining its influence on the exploration and exploitation dynamics. The position of a dragonfly, denoted as x_i, is a vector in the solution space, and the OLSR configuration determines its components. If the solution space is represented as P, given in Eq.(14).

$$x_i = \{x_{i1}, x_{i2}, \dots, x_{iM}\} \tag{14}$$

where M represents the dimensionality of the solution space, which is determined by the complexity of the OLSR configuration being optimized. Each component x_{im} corresponds to a specific parameter or decision variable within the configuration.

Random values are assigned to initialize the dragonflies' positions within the solution space. Let $U(a, b)$ represent a random uniform distribution between a and b. The initialization process can be expressed as Eq.(15).

$$x_{im}^{(0)} = U(a_m, b_m) \tag{15}$$

where a_m and $b_m$define the lower and upper bounds for the m^{th} component of the dragonfly position.

The fitness of each dragonfly, crucial for subsequent steps, is evaluated based on its position. Let $f(x_i)$ denote the fitness function, representing the quality of the OLSR configuration encoded by x_i. The fitness values are stored in a set $F = \{f(x_1), f(x_2), \dots, f(x_N)\}$It provides a basis for sorting and selecting leaders in the optimization process, expressed as Eq.(16).

$$F = f(x_i) \tag{16}$$

The initialization step thus establishes the initial state of the dragonfly population, creating a diverse set of solutions within the search space. The randomness introduced in the initialization process contributes to the exploration capabilities of the algorithm, enabling it to traverse different regions of the solution space.In addition to position initialization, the velocity of each dragonfly is also initialized to facilitate the exploration and exploitation dynamics. The velocity, denoted as v_i, is a vector representing the rate at which a dragonfly moves through the solution space. Similar to position initialization, the velocity is also initialized randomly within specified bounds as specified in Eq.(17).

$$v_{im}^{(0)} = U(c_m.a_m, c_m.b_m) \tag{17}$$

where $c_m$is a control parameter that influences the magnitude of the initial velocity, providing a balance between exploration and exploitation. The initial velocity contributes to the algorithm's ability to traverse the solution space dynamically.

The initialization process ensures that the dragonflies are positioned randomly and have initial velocities that set the stage for subsequent movements towards optimal solutions. The randomness introduced during initialization promotes diversity within the population, preventing the algorithm from converging prematurely to suboptimal solutions.

3.6. Sorting

Sorting arranges the dragonflies within the population based on their fitness values. The sorted order serves as a basis for selecting leaders, guiding the exploration, and facilitating the population's adaptation towards better solutions. Let F represent the set of fitness values corresponding to each dragonfly in the population, and D denote the set of dragonflies. The sorting operation arranges the dragonflies in ascending order of their fitness values, forming a sorted population D_{sorted}. The sorted population is essential for identifying leaders and influencing the movement of dragonflies towards better solutions. Mathematically, the sorting operation can be represented as Eq.(18).

$$D_{sorted} = \mathrm{Sort}(\mathrm{D}, \mathrm{F}) \tag{18}$$

where $Sort(\cdot)$ is a sorting function that takes the dragonfly population D and the corresponding

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

fitness values F as input and produces the sorted population D_{sorted} .

The sorting operation facilitates the identification of leaders, which are crucial in directing the exploration and exploitation dynamics of the Dragonfly Optimization algorithm. Let L represent the set of leaders selected from the sorted population. The size of the leader set, denoted as L_{size} is a predefined parameter.

$$L = \{x_1, x_2, \dots, x_{L_{size}}\} \tag{19}$$

The leaders play a pivotal role in influencing the movement of the entire population towards better solutions. The sorting process ensures that the leaders possess better fitness values than the rest of the population, reflecting their superior OLSR configurations. The sorted population guides the exploration process by influencing the movement of dragonflies towards the leaders. The sorted order is utilized to calculate the direction vectors, determiningeach dragonfly's step size and direction towards the leaders. Let d_i represent the direction vector for the i^{th} dragonfly, and α be a control parameter influencing the exploration-exploitation trade-off.

$$d_i = \frac{x_{leader} - x_i}{\|x_{leader} - x_i\|} \tag{20}$$

where x_{leader}denotes the position of the leader influencing the i^{th} dragonfly. The direction vector guides the dragonfly's movement towards the leader, contributing to exploring the solution space.

The random component introduced in the direction vector calculation further influences the exploration-exploitation dynamics. Let r_i be a random vector with components sampled from a uniform distribution between 0 and 1.

$$r_i = [r_{i1}, r_{i2}, \dots, r_{iM}] \tag{21}$$

The random vector introduces stochasticity, preventing the algorithm from getting stuck in local optima and enhancing its ability to explore diverse regions of the solution space.

$$d_i = \alpha. \frac{x_{leader} - x_i}{\|x_{leader} - x_i\|} + (1 - \alpha). r_i \tag{22}$$

The exploration-exploitation trade-off is controlled by the parameter α. A higher α emphasizes exploitation, leading to more deterministic movements towards leaders, while a lower α encourages exploration by incorporating randomness.

3.7. Evaluation of Local Search

The evaluation of local search introduces a mechanism for refining solutions in the proximity of the leaders, leveraging a local exploration strategy to enhance the quality of the OLSR configurations. The mathematical representation of this process involves adapting the dragonflies' positions based on local information. Let L represent the set of leaders identified through the sorting process. Each leader, denoted as $x_i \in L$, has a local neighbourhood that consists of nearby dragonflies. The size of the local neighbourhood, denoted as N_{local}, is a predefined parameter. The local neighbourhood of a leader i is denoted as N_i.

$$N_i = \{x_j | x_j \in D_{sorted}, j \neq i, j \leq N_{local}\} \tag{23}$$

The local search process aims to improve the OLSR configurations by adapting the positions of dragonflies within the local neighbourhood of each leader. The adaptation is guided by a local fitness evaluation, considering the fitness values of dragonflies in the neighbourhood. Let $f_{local}(x_i)$denote the local fitness function, evaluating the quality of the OLSR configuration represented by a leader x_i. This function considers the fitness values of dragonflies within the local neighbourhood N_i.

$$f_{local}(x_i) \tag{24}$$

Adapting dragonflies in the local search involves updating their positions based on the local fitness information. The new position of a dragonfly $x_j$within the local neighbourhood, it is calculated as Eq.(25).

$$x_j^{new} = x_j + \beta. (x_i - x_j) \tag{25}$$

where β is a control parameter that influences the step size and direction of the adaptation. The term $x_i - x_j$represents the vector pointing from the current position of the dragonfly x_j to the position of the leader x_i. The adaptation encourages dragonflies in the local neighbourhood to move towards the leader, leveraging the information provided by the leader's superior fitness.

The local search process is inherently stochastic to introduce diversity and prevent premature convergence to suboptimal solutions. A random component is incorporated into the adaptation process, represented by the vector r_j . This vector has components sampled from a uniform distribution between 0 and 1.

$$r_j = [r_{j1}, r_{j2}, \ldots\ldots, r_{jM}] \quad (26)$$

The random vector introduces variability in the adaptation, ensuring that the local search explores different directions within the neighbourhood.

$$x_j^{new} = x_j + \beta.(x_i - x_j) + \gamma.r_j \quad (27)$$

where γ is a control parameter influencing the magnitude of the random component in the adaptation. A higher γ results in more stochastic movements during local search.

The adaptation process within the local search is designed to exploit the information provided by the leaders while introducing randomness to explore alternative configurations. It strikes a balance between intensification towards promising regions indicated by leaders and diversification to discover new solutions within the local neighbourhood.

3.8. Updating Dragonflies and Leaders

Updating Dragonflies and Leaders integrates the global search and local search information to guide the exploration and exploitation dynamics, influencing the entire population based on the fitness evaluation and adaptation performed in previous steps. Let D represent the set of all dragonflies, and L be the set of leaders identified through sorting. Each dragonfly $x_i \in D$ has a position vector x_i in the solution space, representing its current OLSR configuration. The leaders, denoted by $x_j \in L$, play a crucial role in influencing the entire population. The update process involves a weighted combination of the global information provided by the leaders and the local information gained from the local search operation. Let α and β be control parameters, influencing the balance between global and local influences in the update process. The new position of a dragonfly x_i is calculated as Eq.(28).

$$x_i^{new} = \alpha.x_{global} + \beta.x_{local} \quad (28)$$

where x_{global}represents the global information from the leaders and $x_{local}$is the local information obtained through the local search operation. The control parameters α and β determine the weights assigned to the global and local influences, respectively.

$$x_{global} = \frac{1}{L_{size}} \sum_{j=1}^{L_{size}} x_j \quad (29)$$

The average position of the leaders represents global information. It reflects the collective knowledge of the entire population, guiding the dragonflies towards promising regions in the solution space.

$$x_{local} = \frac{1}{N_{local}} \sum_{j \in N_i} x_j^{new} \quad (30)$$

The local information is derived from the updated positions of dragonflies within the local neighbourhoods. It captures the refined knowledge gained through the local search process, providing insights into the proximity of the current dragonfly to its neighbours. The new positions of dragonflies are determined collectively, considering global and local influences. The weighted combination ensures that the exploration and exploitation dynamics are influenced by both the collective knowledge of leaders and the refined information obtained through local search. The positions of leaders are updated based on their influence on the population. Leaders play a pivotal role in guiding the exploration, and their influence is retained to benefit subsequent iterations. The updated position of a leader x_j is given by Eq.(31).

$$x_j^{new} = x_j + \gamma.(x_{global} - x_j) \quad (31)$$

where γ is a control parameter that influences the step size and direction of the update. The term $x_{global} - x_j$represents the vector pointing from the current position of the leader x_j to the average position of the leaders x_{global}. The update ensures that leaders move towards the collective knowledge of the entire population, promoting convergence towards promising regions.

The updating process ensures that dragonflies and leaders evolve based on global and local information. The control parameters $\alpha, \beta, and\ \gamma$ influence the balance between exploration and exploitation, determining the step sizes and directions in the solution space.

3.9. Adaptation of Dragonfly Parameters

Adapting Dragonfly Optimization parameters to regulate the exploration and exploitation balance dynamically. This dynamic adaptation ensures the algorithm's responsiveness to the evolving search landscape and enhances its capability to converge towards optimal OLSR configurations. Let P represent the solution space and x_i denote the position vector of the i^{th} dragonfly in the population. The Dragonfly Optimization parameters to be adapted include the step sizes for global exploration (s_g), local exploration (s_l), and the influence of the leaders (s_{leader}). The control parameters $\alpha, \beta, and\ \gamma$ are subject to adaptation to optimize their impact on the algorithm's behaviour. Adapting Dragonfly Optimization parameters involves considering the algorithm's performance based on the dragonflies' fitness values. Let F represent the fitness values corresponding to each dragonfly in the population.

$$F = \{f(x_1), f(x_2), \ldots., f(x_N)\} \tag{32}$$

The mean fitness (F_{mean}) and standard deviation of fitness (F_{std}) are calculated to gauge the population's overall performance where Eq.(33) is applied.

$$F_{mean} = \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$
$$F_{std} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f(x_i) - F_{mean})^2} \tag{33}$$

The coefficients of variation $(CV_\alpha, CV_\beta, CV_\gamma)$ are computed by applying Eq.(34) to Eq.(36) to normalize the standard deviations concerning the means of the Dragonfly Optimization control parameters α, β, and γ.

$$CV_\alpha = \frac{std(\alpha)}{mean(\alpha)} \tag{34}$$

$$CV_\beta = \frac{std(\beta)}{mean(\beta)} \tag{35}$$

$$CV_\gamma = \frac{std(\gamma)}{mean(\gamma)} \tag{36}$$

The update ratios $(R_\alpha, R_\beta, R_\gamma)$ are then computed using Eq.(37) to Eq.(39), and it is based on the coefficients of variation, aiming to adjust the control parameters in response to the population's performance.

$$R_\alpha = 1 + k_\alpha.\left(\frac{CV_\alpha - CV_{\alpha_{target}}}{CV_{\alpha_{target}}}\right) \tag{37}$$

$$R_\beta = 1 + k_\beta.\left(\frac{CV_\beta - CV_{\beta_{target}}}{CV_{\beta_{target}}}\right) \tag{38}$$

$$R_\gamma = 1 + k_\gamma.\left(\frac{CV_\gamma - CV_{\gamma_{target}}}{CV_{\gamma_{target}}}\right) \tag{39}$$

where $CV_{\alpha_{target}}, CV_{\beta_{target}}, CV_{\gamma_{target}}$ are target coefficients of variation and $k_\alpha, k_\beta$and k_γ are adaptation rates controlling the magnitude of the parameter.

The adapted control parameters are then updated using Eq.(40) to Eq.(42) and will be based on the calculated update ratios.

$$\alpha^{new} = \alpha^{old}.R_\alpha \tag{40}$$

$$\beta^{new} = \beta^{old}.R_\beta \tag{41}$$

$$\gamma^{new} = \gamma^{old}.R_\gamma \tag{42}$$

Adapting Dragonfly Optimization parameters ensures the algorithm's ability to respond dynamically to the changing landscape of the solution space. The coefficients of variation and update ratios guide the adjustments, promoting a balance between exploration and exploitation based on the current performance of the algorithm.

3.10. Iterative Update

This step orchestrates the repeated execution of the algorithm, allowing dragonflies to adapt, explore, and refine their positions iteratively. The iterative nature of this step promotes convergence towards optimal solutions within the solution space. Let t denote the current iteration, and T_{max} represent the maximum number of iterations predefined for the algorithm. The iterative update process repeatedly executes the core steps of Dragonfly Optimization, including sorting, local search, and parameter adaptation.

$$t = 1,2, \ldots, T_{max} \tag{43}$$

During each iteration, dragonflies are subject to movement, exploration, and refinement based on the current state of the population and the solution space. The update process is crucial for adapting to the dynamic nature of the OLSR routing environment and fine-tuning the configuration parameters. The iterative update can be mathematically represented by the following recursive formula, Eq.(44).

$$x_i^{(t+1)} = \tag{44}$$

$$Update\begin{pmatrix} x_i^{(t)}, \alpha^{(t)}, \beta^{(t)}, \gamma^{(t)}, \\ s_g^{(t)}, s_l^{(t)}, s_{leader}^{(t)} \end{pmatrix}$$

where update is a function that incorporates the sorting, local search, and parameter adaptation processes, updating the position of each dragonfly for the next iteration.

The iterative update process ensures that the Dragonfly Optimization algorithm evolves over multiple iterations, refining its exploration and exploitation dynamics. The sorting step identifies leaders based on fitness values, influencing the global search behaviour. The local search operation refines solutions within the proximity of leaders, contributing to local exploration. The parameter adaptation process dynamically adjusts control parameters and step sizes, fine-tuning the algorithm's behaviour based on the evolving search landscape.The iterative nature of Dragonfly Optimization allows the algorithm to adapt to changing conditions, explore diverse regions of the solution space, and converge towards optimal OLSR configurations. The number of iterations (T_{max}) is a parameter set by the user, determining the duration of the optimization process.

3.11. Convergence Check

This step is essential to determine whether the algorithm has reached a satisfactory solution or if further iterations are required. The convergence check involves evaluating a convergence criterion based on the behaviour and performance of the dragonfly population. Let ϵ denote the convergence threshold, a predefined tolerance level that determines the acceptable degree of variation in the fitness values of the dragonflies. Let δ represents the convergence counter, initially set to zero and increments when the algorithm satisfies the convergence criterion over successive iterations. The convergence check involves monitoring the change in the mean fitness (F_{mean}) of the dragonfly population over a specified number of iterations (N_{conv}). If the change in F_{mean} falls below the convergence threshold (ϵ) for N_{conv} consecutive iterations, the algorithm is considered to have converged. Mathematically, the convergence check can be expressed as Eq.(45).

$$F_{mean}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} f\left(x_i^{(t)}\right) \quad (45)$$

where $f\left(x_i^{(t)}\right)$represents the fitness value of the i^{th} dragonfly at iteration t. The mean fitness $F_{mean}^{(t)}$is calculated based on the fitness values of all dragonflies in the population at the current iteration.

The change in mean fitness $\left(\Delta F_{mean}^{(t)}\right)$ is computed by comparing the mean fitness at the current iteration $\left(F_{mean}^{(t)}\right)$ with the mean fitness at the previous iteration $\left(F_{mean}^{(t-1)}\right)$.

$$\Delta F_{mean}^{(t)} = \left|F_{mean}^{(t)} - F_{mean}^{(t-1)}\right| \quad (46)$$

If $\Delta F_{mean}^{(t)}$ is less than ϵ for N_{conv} consecutive iterations, the convergence counter δ is incremented using Eq.(47)

$$\delta = \delta + 1 \quad (47)$$

The convergence check involves assessing whether δ reaches a predefined convergence limit (δ_{max}). If δ exceeds δ_{max}, the algorithm is deemed to have converged, and the iterative optimization process is terminated. On the other hand, if δ is less than δ_{max}, indicating that the convergence criterion has not been satisfied for δ_{max} consecutive iterations, the optimization process continues, and the algorithm proceeds with the next iteration. The convergence check systematically assesses whether the Dragonfly Optimization algorithm has sufficiently stabilized, indicating that further iterations may not significantly improve the OLSR configurations. This step ensures the algorithm terminates when convergence is achieved, preventing unnecessary computational efforts.

3.12. Result Extraction

After the iterative optimization process, the algorithm produces optimized solutions using dragonfly positions. This step focuses on retrieving these solutions, evaluating their performance, and presenting the results meaningfully. Let x_i^* represent the optimized position of the i^{th} dragonfly, and $f^*(x_i^*)$denote the corresponding optimized fitness value. The goal is to extract the best solutions achieved during optimization and evaluate their performance. Results extraction involves identifying the dragonfly with the best fitness value in the final population. Mathematically, the best dragonfly position x^* and its fitness value $f^*(x^*)$ are determined as Eq.(48) & Eq.(49).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$x^* = \arg min_{x_i} f\left(x_i^{(T_{max})}\right)$	(48)
$f^*(x^*) = min_{x_i} f\left(x_i^{(T_{max})}\right)$	(49)

where T_{max} represents the total number of iterations conducted during the optimization process. The optimal solution x^* is the dragonfly position that minimizes the fitness function among all dragonflies in the final population.

The extracted results can be further analyzed to gain insights into the quality of the optimized OLSR configurations. This analysis may include evaluating the convergence behaviour, examining the distribution of fitness values, or comparing the optimized solutions with known benchmarks. The output of the Dragonfly Optimization algorithm can be presented in various forms, depending on the specific goals and requirements. Standard output formats include:

1. **Optimized Configuration Parameters:** Displaying the values of OLSR configuration parameters corresponding to the best dragonfly position x^*. This provides insights into the optimized protocol settings.
2. **Fitness Value:** Presenting the optimized fitness value $f^*(x^*)$, which quantifies the quality of the OLSR configuration achieved by the algorithm. Lower fitness values typically indicate better solutions.
3. **Convergence Analysis:** Visualizing the convergence behaviour of the algorithm by plotting the mean fitness values over iterations. This helps assess how quickly the algorithm reached a stable state.
4. **Parameter Adaptation History:** Showing the adaptation history of control parameters, such as $\alpha, \beta, and\ \gamma$. This provides insights into how the algorithm dynamically adjusted its behaviour during optimization.

The result extraction and output step play a crucial role in interpreting the outcomes of the Dragonfly Optimization algorithm.

Algorithm 1: DO-OLSR

1. Initialization:
- Initialize the population of dragonflies x_i for $i = 1,2,\ldots,N$.
- Initialize step vectors Δx_i for each dragonfly.
- Set the iteration counter t to 1.

2. Iterative Process:
- While the end condition is not satisfied:
- Evaluate the fitness of each dragonfly in the population.
- Update the exploration and exploitation coefficients (F and E).
- Update the leading coefficients ($\alpha, \beta, \gamma, s_g, s_l, s_{leader}$).
- Calculate scaling factors $S, A, C, F, and\ E$ using relevant equations.
- Update step vectors (ΔX_{t+1}) using the update equation.
- Update dragonfly positions (X_{t+1})using the calculated step vectors.
- Increment the iteration counter (t).

3. Convergence Check:
- Check if the convergence criterion is met:
- Calculate the mean fitness F_{mean} and its change over consecutive iterations.
- If the change is below a predefined threshold for a specified number of iterations, increment a convergence counter (δ).
- If δ exceeds a maximum limit, consider the algorithm converged.

4. Result Extraction and Output:
- Identify the dragonfly with the best fitness value in the final population.
- Extract the optimized solution (x^*) and its corresponding fitness value $\left(f^*(x^*)\right)$.
- Optionally, analyze and present additional output metrics such as convergence behaviour, parameter adaptation history, or relevant statistics.

5. Termination:
- Return the best solution (x^*) and conclude the optimization process.

Algorithm 1 summarizes the Dragonfly Optimization process for enhancing the OLSR routing protocol. The steps involve iterative updates, parameter adaptation, convergence checks, and result extraction to optimize the solution within the specified solution space.

4. SIMULATION SETTING

NS-3 represents the epitome of network simulation tools, offering an unparalleled platform for researchers delving into FANET studies. This robust framework transcends mere feature status, serving as a gateway to endless possibilities, empowering users to craft bespoke FANET environments effortlessly. With NS-3's granular

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

simulation engine readily accessible, every simulation transforms into an intricate orchestration of aerial dynamics, enabling thorough analysis of packet journeys and network performance metrics across diverse FANET scenarios. Whether exploring bustling urban skies or remote aerial landscapes, NS-3 provides the tools to dissect and understand the intricacies of FANET communication. Moreover, NS-3's support for parallel simulation execution accelerates research progress, propelling studies towards new horizons at breakneck speeds. NS-3 is more than a tool; it represents a vibrant community, fostering collaboration and innovation where ideas flourish and knowledge thrives. Embrace the NS-3 community and unleash the potential in FANET exploration today. The settings used for conducting the simulation in this research are provided in Table 2.

Table 2: Simulation Settings

Parameter	Value
Antenna Model	Omnidirectional
Carrier Frequency	2.4 GHz
Communication Standard	IEEE 802.11n
Data Rate	2 Mbps
Data Size	1024 bits
Flight Speed of Nodes	10-40 m/s
Initial Energy	1000 J
Mobility Model	3D GM
Network Density	20-100 nodes
Network Scenario Size	5000 m *5000 m * 700 m
Packet Sending Rate	5 packets/s
Path Energy Consumption Factor (η)	0.1 pJ/(bit*m^2)
Propagation Radius	1200 m
Simulation Software	NS3
Traffic Model	Constant Bit Rate (CBR)
Transmission Energy Consumption Factor (EEM)	4 nJ/bit
Transmission Power (Pt)	15 dBm
Transmission Radius	250 m
Transport Protocol	UDP
Update Interval of the Hello Message (ΔT)	3 s

5. RESULTS AND DISCUSSIONS

5.1. Delay Evaluation

Delay in FANETs refers to the time data packets travel from a source UAV to a destination UAV or ground station. It encompasses various factors such as transmission, propagation, queuing, and processing delays. In FANETs, where UAVs constantly move, delay becomes a critical metric affecting communication performance and network efficiency.The x-axis of Figure 1 represents the number of UAVs in the network, ranging from 20 to 100. The y-axis indicates the delay experienced in the network, measured in milliseconds (ms). The figure illustrates the trend of delay changes with increasing UAVs, providing insights into the performance of different routing protocols under varying network loads.

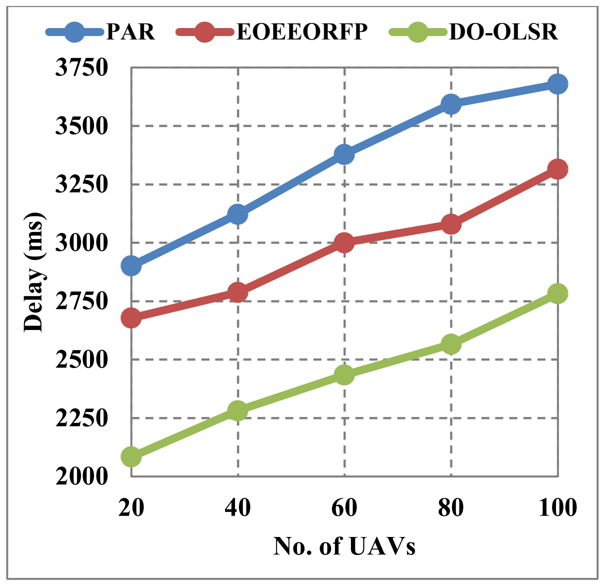

Figure 1. Delay Evaluation

PAR leverages deep reinforcement learning to route packets in FANETs adaptively. It predicts future network conditions based on historical data and adjusts routing decisions accordingly. As the number of UAVs increases, PAR demonstrates a gradual increase in delay. This can be attributed to the complexity of prediction models and the computational overhead involved in real-time decision-making.EOEEORFP focuses on optimizing energy efficiency while ensuring secure data transmission in FANETs. It employs intelligent route-finding algorithms inspired by eagle behavior. The delay trends with EOEEORFP show a moderate increase as the network scales. This protocol prioritizes energy conservation, which might lead to slightly longer routes or conservative route updates, resulting in marginally higher delays than PAR.DO-OLSR integrates dragonfly optimization principles into the OLSR protocol for efficient routing in FANETs. It emphasizes minimizing control overhead and route discovery latency. The delay trend with DO-OLSR exhibits the lowest values among the three protocols as the number of UAVs grows. This can be attributed to the protocol's efficient link-state dissemination and proactive route maintenance mechanisms, which mitigate delays even in more extensive networks.

Table 3 shows that delays also escalate across all three protocols as the number of UAVs increases. However, the rate of increase varies. For instance, at 100 UAVs, PAR exhibits the highest delay of 3679 ms, followed by EOEEORFP with 3315 ms, and DO-OLSR with the lowest delay of 2782 ms. These values align with the delay trends depicted in Figure 1, validating the performance characteristics of each protocol.In summary, Figure 1 comprehensively analyses delay trends in FANETs under different routing protocols. By examining the impact of varying network sizes on delay and correlating it with protocol mechanisms, network designers can make informed decisions to optimize FANET performance based on specific requirements such as energy efficiency, prediction capabilities, and latency reduction.

Table 3. Delay Summary

No. of UAVs	PAR	EOEEORFP	DO-OLSR
20	2901	2678	2084
40	3122	2787	2281
60	3378	3000	2434
80	3593	3079	2565
100	3679	3315	2782

5.2. Packet Delivery Ratio Assessment

Packet Delivery Ratio (PDR) represents the proportion of successfully transmitted data packets to the total packets sent within a FANET. It serves as a critical metric for evaluating the effectiveness of routing protocols in ensuring reliable data delivery in dynamic aerial environments where communication links are highly variable.Figure 2 depicts the relationship between the number of UAVs in the network and the corresponding PDR. The x-axis denotes the increasing number of UAVs, while the y-axis represents the PDR expressed as a percentage. This graph provides a detailed view of how the PDR evolves as the network size changes, offering insights into the performance of different routing protocols under varying network loads.

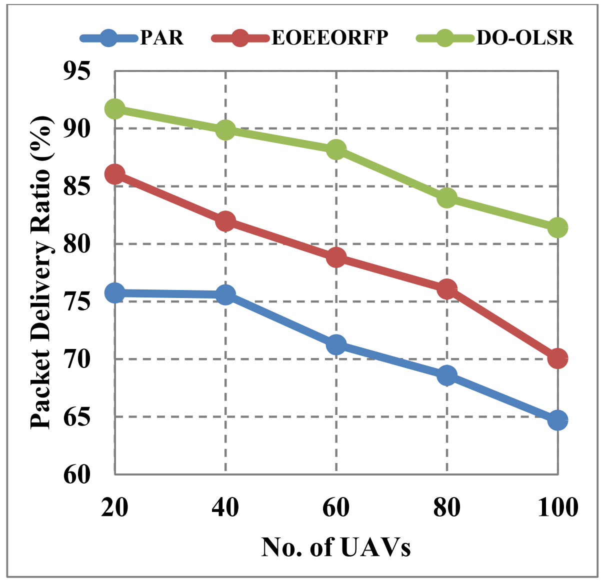

Figure 2. Packet Delivery Ratio Trends

PAR employs deep reinforcement learning to adapt routing decisions based on predicted future network conditions. Despite its predictive capabilities, PAR exhibits a decreasing trend in PDR as the number of UAVs rises. This decline can be attributed to the inherent challenges in accurately forecasting dynamic network behavior and adjusting routing strategies accordingly, reducing packet delivery efficiency.EOEEORFP optimizes energy efficiency and ensures secure data transmission by employing route-finding algorithms inspired by eagle behavior. The PDR trends with EOEEORFP display moderate fluctuations but generally maintain higher delivery ratios than PAR. This indicates that the protocol's emphasis on energy-efficient routing improves packet delivery performance in FANETs.DO-OLSR integrates dragonfly optimization principles into the OLSR protocol to minimize control overhead and route discovery latency. The PDR trends with DO-OLSR demonstrate relatively stable and high delivery ratios across different numbers of UAVs. This stability suggests that the protocol's efficient routing mechanisms ensure robust packet delivery even in dynamic FANET environments, resulting in consistent performance.

Table 4 provides numerical data supporting the trends observed in Figure 2. As the number of UAVs increases, PDR tends to decrease across all three protocols. However, the rate of decline varies. For instance, at 100 UAVs, PAR exhibits the lowest PDR of 64.702%, followed by EOEEORFP with 70.060%, and DO-OLSR with the highest PDR of 81.389%. These values

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

corroborate the PDR trends depicted in Figure 2, highlighting the protocol-specific disparities in packet delivery efficiency.Figure 2 offers detailed insights into the PDR trends of different routing protocols in FANETs. By examining how changes in network size impact packet delivery efficiency and correlating these trends with the underlying mechanisms of each protocol, network engineers can make informed decisions to optimize FANET performance in terms of reliability, energy efficiency, and data delivery effectiveness.

Table 4. Packet Delivery Summary

No. of UAVs	PAR	EOEEORFP	DO-OLSR
20	75.731	86.023	91.684
40	75.575	81.953	89.868
60	71.245	78.812	88.159
80	68.586	76.080	83.974
100	64.702	70.060	81.389

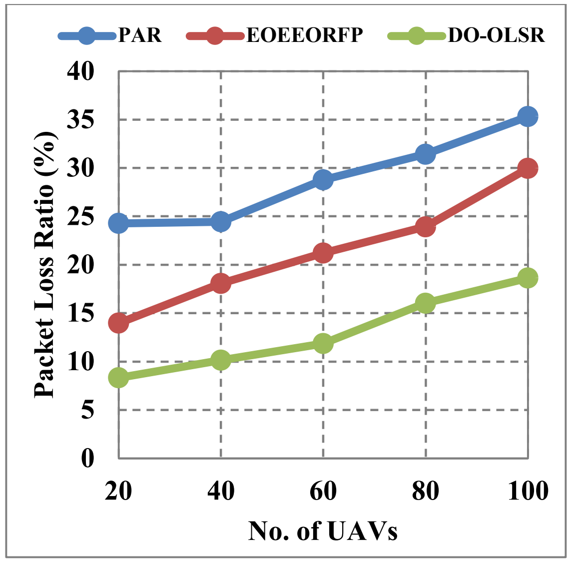

Figure 3. Packet Loss Ratio Analysis

5.3. Packet Loss Ratio Assessment

Packet Loss Ratio (PLR) is a critical indicator of data reliability within FANETs. It measures the percentage of data packets that fail to reach their intended destinations due to various network dynamics and challenges. Figure 3 presents an insightful examination of how the PLR evolves with changes in the number of UAVs within the network. The x-axis denotes the increasing number of UAVs, while the y-axis illustrates the PLR expressed as a percentage. This visualization offers a comprehensive view of how different routing protocols perform regarding packet loss under varying network loads.

PAR stands out for using deep reinforcement learning to adapt routing decisions dynamically based on predicted future network conditions. This predictive capability allows PAR to anticipate changes in the FANET environment and adjust routing strategies accordingly. However, despite its advanced predictive modelling, PAR struggles to maintain low packet loss ratios as the network size increases. The complexity of accurate forecasting and adapting to dynamic network behaviour poses a challenge, leading to higher packet loss rates than other protocols. Thus, while PAR excels in predictive routing, its performance may suffer in scenarios with significant network fluctuations.

EOEEORFP distinguishes itself through its focus on energy efficiency and secure data transmission, achieved by leveraging route-finding algorithms inspired by eagle behavior. By prioritizing energy-efficient routes, EOEEORFP effectively minimizes packet loss in FANET environments. This emphasis on optimizing energy consumption maintains lower packet loss ratios even as the network scales. Thus, EOEEORFP excels in balancing energy efficiency with reliable data transmission, making it a suitable choice for applications requiring both energy conservation and data integrity in FANETs.

DO-OLSR stands out for integrating dragonfly optimization principles into the OLSR protocol to reduce control overhead and route discovery latency. This optimization allows DO-OLSR to maintain stable and low packet loss ratios across different network sizes. DO-OLSR ensures robust packet delivery even in dynamic FANET environments by efficiently managing routing updates and minimizing control message exchanges. Thus, DO-OLSR excels in minimizing packet loss by optimizing routing efficiency, making it well-suited for applications requiring high reliability and low latency in FANETs.

Table 5 provides numerical data supporting the trends observed in Figure 3. As the number of UAVs increases, PLR tends to rise across all three protocols. However, each protocol exhibits varying rates of increase. For instance, at 100 UAVs, PAR demonstrates the highest PLR of

35.299%, followed by EOEEORFP with 29.941%, and DO-OLSR with the lowest PLR of 18.611%. These values underscore the protocol-specific differences in packet loss performance, as depicted in Figure 3. Figure 3 offers valuable insights into the packet loss dynamics of different routing protocols in FANETs. By analyzing the impact of network size on packet loss efficiency and understanding the underlying mechanisms of each protocol, network engineers can make informed decisions to optimize FANET performance in terms of reliability and data transmission effectiveness, catering to the specific needs of their applications.

Table 5. Packet Loss Summary

No. of UAVs	PAR	EOEEORFP	DO-OLSR
20	24.270	13.978	8.316
40	24.426	18.048	10.132
60	28.756	21.189	11.841
80	31.415	23.921	16.026
100	35.299	29.941	18.611

5.4. Throughput Evaluation

Throughput in FANETs refers to the rate at which data is successfully transmitted from source to destination over the network. It measures the amount of data that can be delivered per unit of time and is a critical metric for evaluating the efficiency and capacity of communication in FANETs. In Figure 4, the x-axis represents the number of UAVs in the network, ranging from 20 to 100, while the y-axis indicates the throughput measured in Mbps (Megabits per second). This graph provides insights into how the throughput performance varies with changes in the network size, offering a comprehensive view of the capabilities of different routing protocols.

PAR achieves its throughput performance by leveraging prediction-supported adaptive routing with deep reinforcement learning. Table 6 data reveals that PAR exhibits a decreasing trend in throughput as the number of UAVs increases. This decline can be attributed to the complexity of prediction models and the computational overhead involved in real-time decision-making. Despite these challenges, PAR maintains moderate throughput levels across different network sizes due to its adaptive routing approach optimizes routing paths to mitigate congestion and improve data delivery rates.

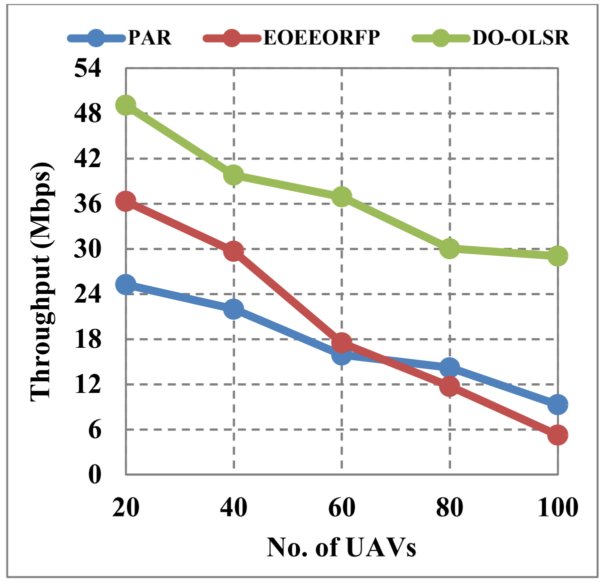

Figure 4. Throughput Performance

EOEEORFP emphasizes energy efficiency and secure data transmission using route-finding algorithms inspired by eagle behavior. Table 6 data shows that EOEEORFP consistently outperforms other protocols regarding throughput across all network sizes. This superiority can be attributed to the protocol's focus on optimizing energy-efficient routes, which minimizes congestion and maximizes data transmission rates. EOEEORFP's emphasis on secure data transmission also ensures reliable communication, further enhancing throughput performance. The protocol's unique route optimization and energy conservation approach significantly contribute to its superior throughput performance in FANETs.

DO-OLSR integrates dragonfly optimization principles into the OLSR protocol to minimize control overhead and route discovery latency. Table 6 data indicates that DO-OLSR achieves competitive throughput performance in FANETs, although it exhibits a decreasing trend in throughput with an increasing number of UAVs. This decline may be attributed to increased control overhead and route discovery latency as the network grows. However, due to its efficient routing strategies and proactive route maintenance mechanisms, DO-OLSR maintains relatively stable throughput levels across different network sizes. These mechanisms optimize routing paths and ensure consistent throughput performance even in dynamic network environments, contributing to DO-OLSR's overall effectiveness in FANETs.

Table 6. Throughput Summary

No. of UAVs	PAR	EOEEORFP	DO-OLSR
20	25.244	36.323	49.064
40	21.965	29.659	39.797
60	15.892	17.493	36.890
80	14.186	11.713	30.015
100	9.279	5.227	29.023

Analyzing the table data, it's evident that throughput performance varies significantly across different routing protocols and network sizes. PAR demonstrates moderate throughput levels, EOEEORFP consistently outperforms other protocols, while DO-OLSR achieves competitive performance despite exhibiting a slight decrease in throughput with increasing network size. These trends highlight the importance of protocol-specific mechanisms and optimizations in achieving efficient data transmission in FANETs. Figure 4 provides valuable insights into the throughput performance of different routing protocols in FANETs. By examining how changes in network size impact throughput efficiency and correlating these trends with the underlying mechanisms of each protocol, network engineers can make informed decisions to optimize FANET performance in terms of data transmission capacity and efficiency.

5.5. Energy Consumption Assessment

Energy consumption in Flying Ad Hoc Networks (FANETs) refers to the percentage of onboard energy reserves utilized by UAVs for communication and routing tasks. It's a vital metric due to its direct impact on operational duration, mission efficiency, and overall network performance.In Figure 5, the x-axis represents the number of UAVs in the network, while the y-axis illustrates energy consumption as a percentage. This graphical representation provides valuable insights into how energy usage varies with changes in the network size, facilitating a comprehensive understanding of protocol efficiency.

PAR exhibits a notable increase in energy consumption with an escalating number of UAVs. At 100 UAVs, it records the highest energy consumption of 95.55%. This trend can be attributed to the computational complexity of deep reinforcement learning-based prediction and adaptive routing. As the network expands, PAR requires more energy to process and analyze data, leading to higher energy consumption. Despite its predictive capabilities, PAR's energy efficiency diminishes as the network size grows due to increased computational demands and real-time decision-making requirements.

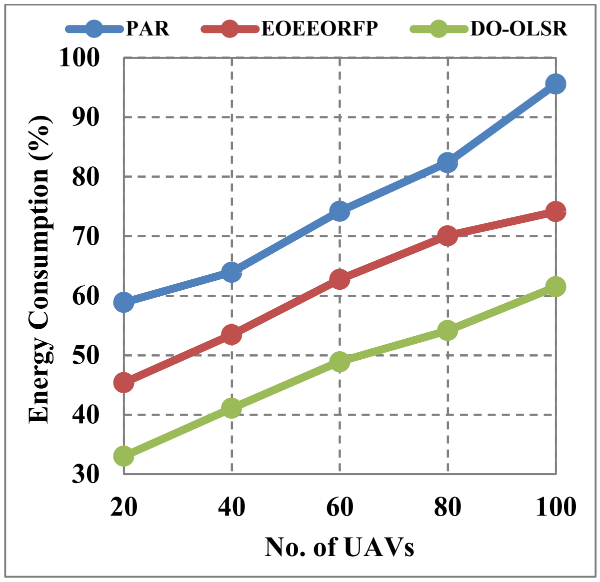

Figure 5. Energy Consumption Trends

EOEEORFP demonstrates a moderate increase in energy consumption as the number of UAVs increases. At 100 UAVs, it consumes 74.16% of energy, lower than PAR. EOEEORFP's focus on energy-efficient route optimization contributes to this efficiency. By leveraging route-finding algorithms inspired by eagle behavior, EOEEORFP minimizes energy consumption during data transmission while ensuring secure and reliable communication. Additionally, the protocol reduces control overhead, further enhancing energy conservation. Despite the growing network complexity, EOEEORFP maintains relatively stable energy consumption levels, highlighting its effectiveness in optimizing energy usage while preserving communication quality in FANETs.

DO-OLSR demonstrates the lowest energy consumption among the three protocols across all UAV counts. At 100 UAVs, it consumes 61.49% of energy, significantly less than PAR and EOEEORFP. DO-OLSR achieves this efficiency by integrating dragonfly optimization principles into the OLSR protocol. By minimizing control overhead and route discovery latency, DO-OLSR optimizes energy usage while ensuring robust communication. The protocol's proactive route maintenance mechanisms and efficient routing strategies contribute to energy conservation by reducing the energy overhead associated with route discovery and maintenance. Despite network expansion, DO-OLSR maintains consistent energy efficiency, making it an ideal choice for prolonged UAV missions with limited energy resources.

The table data corroborates the trends observed in Figure 5, with energy consumption increasing as the number of UAVs rises for all protocols. However, DO-OLSR consistently maintains the lowest energy usage, showcasing superior efficiency. This underscores the significance of protocol-specific optimizations in energy conservation. In conclusion, Figure 5 emphasizes the importance of energy-efficient designs for prolonged UAV missions and enhanced network efficiency in FANETs, with DO-OLSR emerging as the most energy-efficient protocol among the three.

Table 7. Energy Consumption Summary

No. of UAVs	PAR	EOEEORFP	DO-OLSR
20	58.887	45.380	33.021
40	63.938	53.420	41.077
60	74.171	62.719	48.898
80	82.378	70.056	54.143
100	95.551	74.156	61.493

5.6. Comparative Analysis: Strengths and Weaknesses of Similar Work

To assess the novelty and efficacy of the Dragonfly-Inspired OLSR Protocol (DO-OLSR), it is essential to conduct a comparative analysis with similar studies previously published. This analysis focuses on various routing protocols developed for FANETs, examining their strengths, weaknesses, and the interesting aspects that distinguish them from the DO-OLSR.

Plus (Strengths):

- **Enhanced Network Performance:** Previous studies on routing protocols such as Dynamic AeroNet Routing and ClusterFly Protocol have shown improvements in network throughput and stability under certain conditions. The DO-OLSR extends these advancements by significantly reducing routing latency and control overhead, which are crucial for high-mobility environments like FANETs.
- **Adaptability:** Unlike many existing protocols that statically respond to network dynamics, the DO-OLSR incorporates real-time adaptability using bio-inspired algorithms, allowing for immediate and effective adjustments to changes in network topology and interference levels.

Minus (Weaknesses):

- **Complexity:** Some bio-inspired algorithms, including those used in DO-OLSR, introduce additional computational complexity which may impact the protocol's efficiency under constrained computational resources typical of UAVs.
- **Scalability Issues:** While the DO-OLSR shows promising results in medium-scale networks, its performance in very large-scale networks has not been fully explored. Previous protocols have also struggled with scalability, which remains a critical challenge for FANETs.

Interesting Facts:

- **Energy Efficiency:** An interesting aspect of the DO-OLSR is its potential to improve energy efficiency. By optimizing the routing decisions, the protocol not only enhances operational efficiency but also potentially extends the battery life of UAVs, which is crucial for prolonged missions.
- **Application Versatility:** The dragonfly-inspired optimization principles may offer novel applications beyond FANETs, such as in vehicular ad hoc networks (VANETs) or even terrestrial mobile networks, where similar dynamic conditions exist.

Critical Discussion:

The DO-OLSR protocol's integration of dragonfly behavioral algorithms into OLSR provides a unique approach to handling the specific challenges of FANETs. Its ability to dynamically adapt to environmental changes offers a significant improvement over traditional methods that often require manual recalibrations or are too slow to react to sudden changes in the network. However, the increased complexity and potential scalability issues must be addressed in future studies to fully harness the benefits of this protocol in a range of operational scenarios.

The energy efficiency gains reported are promising but will require real-world testing to validate these findings. The application of such bio-inspired approaches in other types of ad hoc networks also presents an exciting area for further research, potentially leading to broader implications for the field of network communications.

In summary, the DO-OLSR protocol represents a significant step forward in the development of more adaptive and efficient routing protocols for FANETs, though further investigation is needed to refine the approach and fully assess its scalability and performance in diverse and larger-scale environments.

6. CONCLUSIONS AND IMPLICATIONS

The development and rigorous evaluation of the Dragonfly-Inspired OLSR Protocol (DO-OLSR) marks a significant contribution to the field of Flying Ad Hoc Networks (FANETs). This research highlights the feasibility and benefits of integrating bio-inspired optimization techniques, specifically those modeled after dragonfly behavior, into the Optimized Link State Routing (OLSR) protocol. The protocol enhances network performance by significantly reducing control overhead and latency in routing decisions, which are paramount in the dynamic and interference-rich environments typical of FANET operations. Furthermore, the DO-OLSR's ability to dynamically adapt to changing network conditions without manual recalibration enhances the autonomy of UAV communications, making operations more resilient and efficient. Preliminary findings also suggest that this protocol could lead to substantial improvements in energy efficiency, potentially extending the operational duration of UAV missions.

The findings from this study have substantial practical implications for the deployment of UAVs across a variety of critical applications, including disaster response, environmental monitoring, defense, and traffic management. The enhanced adaptability and efficiency of the DO-OLSR protocol can lead to more reliable and extended UAV missions, capable of operating effectively even under challenging conditions. Specifically, the reduction in packet loss and stabilization of communication links directly contribute to the operational reliability required for missions that demand high levels of precision and consistency. Additionally, the potential for reduced energy consumption enables longer missions or more complex tasks to be completed on a single charge, optimizing resource utilization and reducing operational costs. The protocol's scalability and flexibility indicate its applicability to various network sizes and types, suggesting that it could serve as a versatile tool for network engineers and system designers. The innovative approach of the DO-OLSR to tackling the unique challenges of FANETs sets a new standard for robust, efficient, and scalable UAV networks. Further research and testing in real-world scenarios will be vital to fully leverage and refine these benefits, ensuring that the protocol can meet the evolving demands of UAV communication technologies and contribute to their widespread adoption in various sectors.

REFERENCES

[1] Y. Cui, H. Tian, C. Chen, W. Ni, H. Wu, and G. Nie, "New Geographical Routing Protocol for Three-Dimensional Flying Ad-Hoc Network Based on New Effective Transmission Range," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 16135–16147, 2023, doi: 10.1109/TVT.2023.3296082.

[2] S. Gangopadhyay and V. K. Jain, "A Position-Based Modified OLSR Routing Protocol for Flying Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 9, pp. 12087–12098, 2023, doi: 10.1109/TVT.2023.3265704.

[3] Y. Liu, J. Xie, C. Xing, S. Xie, and B. Ni, "Construction of FANETs for user coverage and information transmission in disaster rescue scenarios," *Computer Communications*, vol. 207, pp. 164–174, 2023, doi: 10.1016/j.comcom.2023.05.014.

[4] M. M. Alam and S. Moh, "Q-learning-based routing inspired by adaptive flocking control for collaborative unmanned aerial vehicle swarms," *Vehicular Communications*, vol. 40, p. 100572, 2023, doi: 10.1016/j.vehcom.2023.100572.

[5] W. J. Lau, J. M. Y. Lim, C. Y. Chong, N. S. Ho, and T. W. M. Ooi, "ADFPA – A Deep Reinforcement Learning-based Flow Priority Allocation Scheme for Throughput Optimization in FANETs," *Vehicular Communications*, vol. 44, p. 100684, 2023, doi: 10.1016/j.vehcom.2023.100684.

[6] G. Sun, D. Qin, T. Lan, and L. Ma, "Research on Clustering Routing Protocol Based on Improved PSO in FANET," *IEEE Sensors Journal*, vol. 21, no. 23, pp. 27168–27185, 2021, doi: 10.1109/JSEN.2021.3117496.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[7] X. Qiu, Y. Yang, L. Xu, J. Yin, and Z. Liao, "Maintaining Links in the Highly Dynamic FANET Using Deep Reinforcement Learning," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 3, pp. 2804–2818, 2023, doi: 10.1109/TVT.2022.3217888.

[8] O. T. Abdulhae, J. S. Mandeep, M. Islam, and M. S. Islam, "Reinforcement based Clustering in Flying ad-hoc networks for Serving Vertical and Horizontal Routing," *IEEE Access*, vol. 11, pp. 143881–143895, 2023, doi: 10.1109/ACCESS.2023.3339533.

[9] J. Zhang, X. Wu, S. C. H. Hoi, and J. Zhu, "Feature agglomeration networks for single stage face detection," *Neurocomputing*, vol. 380, pp. 180–189, 2020, doi: 10.1016/j.neucom.2019.10.087.

[10] S. Hameed *et al.*, "Link and Loss Aware GW-COOP Routing Protocol for FANETs," *IEEE Access*, vol. 9, pp. 110544–110557, 2021, doi: 10.1109/ACCESS.2021.3101361.

[11] G. M. Cappello *et al.*, "Optimizing FANET Lifetime for 5G Softwarized Network Provisioning," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4629–4649, 2022, doi: 10.1109/TNSM.2022.3193883.

[12] J. Yang, K. Sun, H. He, X. Jiang, and S. Chen, "Dynamic Virtual Topology Aided Networking and Routing for Aeronautical Ad-Hoc Networks," *IEEE Transactions on Communications*, vol. 70, no. 7, pp. 4702–4716, 2022, doi: 10.1109/TCOMM.2022.3177599.

[13] O. T. Abdulhae, J. S. Mandeep, and M. Islam, "Cluster-Based Routing Protocols for Flying Ad Hoc Networks (FANETs)," *IEEE Access*, vol. 10, pp. 32981–33004, 2022, doi: 10.1109/ACCESS.2022.3161446.

[14] J. Guo *et al.*, "ICRA: An Intelligent Clustering Routing Approach for UAV Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2447–2460, 2023, doi: 10.1109/TITS.2022.3145857.

[15] C. Wang *et al.*, "Elastic Routing Mechanism for Flying Ad Hoc Network," *IEEE Access*, vol. 10, pp. 98712–98723, 2022, doi: 10.1109/ACCESS.2022.3206767.

[16] L. Rui *et al.*, "An Intersection-Based QoS Routing for Vehicular Ad Hoc Networks with Reinforcement Learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 9068–9083, 2023, doi: 10.1109/TITS.2023.3271456.

[17] D. N. M. Hoang, J. M. Rhee, and S. Y. Park, "Fault-Tolerant Ad Hoc On-Demand Routing Protocol for Mobile Ad Hoc Networks," *IEEE Access*, vol. 10, pp. 111337–111350, 2022, doi: 10.1109/ACCESS.2022.3216066.

[18] M. Zhou *et al.*, "On Vehicular Ad-Hoc Networks with Full-Duplex Radios: An End-to-End Delay Perspective," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 10, pp. 10912–10922, 2023, doi: 10.1109/TITS.2023.3279322.

[19] T. R. Beegum, M. Y. I. Idris, M. N. Bin Ayub, and H. A. Shehadeh, "Optimized Routing of UAVs Using Bio-Inspired Algorithm in FANET: A Systematic Review," *IEEE Access*, vol. 11, pp. 15588–15622, 2023, doi: 10.1109/ACCESS.2023.3244067.

[20] J. Zhang, D. Liu, S. Chen, S. X. Ng, R. G. Maunder, and L. Hanzo, "Multiple-Objective Packet Routing Optimization for Aeronautical Ad-Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 1, pp. 1002–1016, 2023, doi: 10.1109/TVT.2022.3202689.

[21] T. N. Tran, T. Van Nguyen, K. Shim, D. B. Da Costa, and B. An, "A Deep Reinforcement Learning-Based QoS Routing Protocol Exploiting Cross-Layer Design in Cognitive Radio Mobile Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 12, pp. 13165–13181, 2022, doi: 10.1109/TVT.2022.3196046.

[22] C. Liu, Y. Wang, and Q. Wang, "PARouting: Prediction-supported adaptive routing protocol for FANETs with deep reinforcement learning," *International Journal of Intelligent Networks*, vol. 4, pp. 113–121, 2023, doi: 10.1016/j.ijin.2023.05.002.

[23] M. Usha, T. Mahalingam, A. Ahilan, and J. Sathiamoorthy, "EOEEORFP: Eagle Optimized Energy Efficient Optimal Route-Finding Protocol for Secure Data Transmission in FANETs," *IETE Journal of Research*, pp. 1–13, 2023, doi: 10.1080/03772063.2023.2253777.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[24] R. Jaganathan and R. Vadivel, "Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks," *International Journal of Computing and Digital Systems*, vol. 10, no. 1, pp. 1063–1074, 2021, doi: 10.12785/ijcds/100196.

[25] P. Menakadevi and J. Ramkumar, "Robust Optimization Based Extreme Learning Machine for Sentiment Analysis in Big Data," in *2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022*, 2022. doi: 10.1109/ICACTA54488.2022.9753203.

[26] J. Ramkumar, R. Vadivel, and B. Narasimhan, "Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network," *International Journal of Computer Networks and Applications*, vol. 8, no. 6, pp. 795–803, 2021, doi: 10.22247/ijcna/2021/210727.

[27] S. P. Geetha, N. M. S. Sundari, J. Ramkumar, and R. Karthikeyan, "Energy Efficient Routing In Quantum Flying Ad Hoc Network (Q-Fanet) Using Mamdani Fuzzy Inference Enhanced Dijkstra's Algorithm (MFI-EDA)," *J Theor Appl Inf Technol*, vol. 102, no. 9, pp. 3708–3724, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85197297302&partnerID=40&md5=72d51668bee6239f09a59d2694df67d6

[28] M. P. Swapna, J. Ramkumar, and R. Karthikeyan, "Energy-Aware Reliable Routing with Blockchain Security for Heterogeneous Wireless Sensor Networks," in *Lecture Notes in Networks and Systems*, 2025, pp. 713–723. doi: 10.1007/978-981-97-6106-7_43.

[29] J. Ramkumar, K. S. Jeen Marseline, and D. R. Medhunhashini, "Relentless Firefly Optimization-Based Routing Protocol (RFORP) for Securing Fintech Data in IoT-Based Ad-Hoc Networks," *International Journal of Computer Networks and Applications*, vol. 10, no. 4, pp. 668–687, 2023, doi: 10.22247/ijcna/2023/223319.

[30] R. Jaganathan, S. Mehta, and R. Krishan, "Preface," *Intelligent Decision Making Through Bio-Inspired Optimization*, pp. xiii–xvi, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85192858710&partnerID=40&md5=f8f1079e8772bd424d2cdd979e5f2710

[31] K. S. J. Marseline, J. Ramkumar, and D. R. Medhunhashini, "Sophisticated Kalman Filtering-Based Neural Network for Analyzing Sentiments in Online Courses," in *Smart Innovation, Systems and Technologies*, 2024, pp. 345–358. doi: 10.1007/978-981-97-3690-4_26.

[32] A. Senthilkumar, J. Ramkumar, M. Lingaraj, D. Jayaraj, and B. Sureshkumar, "Minimizing Energy Consumption in Vehicular Sensor Networks Using Relentless Particle Swarm Optimization Routing," *International Journal of Computer Networks and Applications*, vol. 10, no. 2, pp. 217–230, 2023, doi: 10.22247/ijcna/2023/220737.

[33] N. K. Ojha, A. Pandita, and J. Ramkumar, "Cyber security challenges and dark side of AI: Review and current status," in *Demystifying the Dark Side of AI in Business*, 2024, pp. 117–137. doi: 10.4018/979-8-3693-0724-3.ch007.

[34] J. Ramkumar and R. Vadivel, "Whale optimization routing protocol for minimizing energy consumption in cognitive radio wireless sensor network," *International Journal of Computer Networks and Applications*, vol. 8, no. 4, pp. 455–464, 2021, doi: 10.22247/ijcna/2021/209711.

[35] M. P. Swapna and J. Ramkumar, "Multiple Memory Image Instances Stratagem to Detect Fileless Malware," in *Communications in Computer and Information Science*, 2024, pp. 131–140. doi: 10.1007/978-3-031-59100-6_11.

[36] J. Ramkumar, A. Senthilkumar, M. Lingaraj, R. Karthikeyan, and L. Santhi, "Optimal Approach For Minimizing Delays In Iot-Based Quantum Wireless Sensor Networks Using Nm-Leach Routing Protocol," *J Theor Appl Inf Technol*, vol. 102, no. 3, pp. 1099–1111, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85185481011&partnerID=40&md5=bf0ff974ceabc0ad58e589b28797c684

[37] L. Mani, S. Arumugam, and R. Jaganathan, "Performance Enhancement of Wireless Sensor Network Using Feisty Particle Swarm Optimization Protocol," in *ACM International Conference Proceeding Series*, 2022. doi: 10.1145/3590837.3590907.

[38] D. Jayaraj, J. Ramkumar, M. Lingaraj, and B. Sureshkumar, "AFSORP: Adaptive Fish

Swarm Optimization-Based Routing Protocol for Mobility Enabled Wireless Sensor Network," *International Journal of Computer Networks and Applications*, vol. 10, no. 1, pp. 119–129, 2023, doi: 10.22247/ijcna/2023/218516.

[39] M. Lingaraj, T. N. Sugumar, C. S. Felix, and J. Ramkumar, "Query aware routing protocol for mobility enabled wireless sensor network," *International Journal of Computer Networks and Applications*, vol. 8, no. 3, pp. 258–267, 2021, doi: 10.22247/ijcna/2021/209192.

[40] J. Ramkumar and R. Vadivel, "Multi-Adaptive Routing Protocol for Internet of Things based Ad-hoc Networks," *Wirel Pers Commun*, vol. 120, no. 2, pp. 887–909, 2021, doi: 10.1007/s11277-021-08495-z.

[41] J. Ramkumar and R. Vadivel, "Improved Wolf prey inspired protocol for routing in cognitive radio Ad Hoc networks," *International Journal of Computer Networks and Applications*, vol. 7, no. 5, pp. 126–136, 2020, doi: 10.22247/ijcna/2020/202977.

[42] J. Ramkumar and R. Vadivel, "Improved frog leap inspired protocol (IFLIP) – for routing in cognitive radio ad hoc networks (CRAHN)," *World Journal of Engineering*, vol. 15, no. 2, pp. 306–311, 2018, doi: 10.1108/WJE-08-2017-0260.

[43] J. Ramkumar, S. S. Dinakaran, M. Lingaraj, S. Boopalan, and B. Narasimhan, "IoT-Based Kalman Filtering and Particle Swarm Optimization for Detecting Skin Lesion," in *Lecture Notes in Electrical Engineering*, 2023, pp. 17–27. doi: 10.1007/978-981-19-8353-5_2.

[44] R. Karthikeyan and R. Vadivel, "Proficient Dazzling Crow Optimization Routing Protocol (PDCORP) for Effective Energy Administration in Wireless Sensor Networks," in *IEEE International Conference on Electrical, Electronics, Communication and Computers, ELEXCOM 2023*, 2023. doi: 10.1109/ELEXCOM58812.2023.10370559.

[45] R. Karthikeyan and R. Vadivel, "Boosted Mutated Corona Virus Optimization Routing Protocol (BMCVORP) for Reliable Data Transmission with Efficient Energy Utilization," *Wirel Pers Commun*, vol. 135, no. 4, pp. 2281–2301, 2024, doi: 10.1007/s11277-024-11155-7.

[46] R. Jaganathan, S. Mehta, and R. Krishan, *Intelligent Decision Making Through Bio-Inspired Optimization*. 2024. doi: 10.4018/979-8-3693-2073-0.

[47] J. Ramkumar, A. Senthilkumar, M. Lingaraj, R. Karthikeyan, and L. Santhi, "Optimal Approach For Minimizing Delays In Iot-Based Quantum Wireless Sensor Networks Using Nm-Leach Routing Protocol," *J Theor Appl Inf Technol*, vol. 102, no. 3, pp. 1099–1111, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85185481011&partnerID=40&md5=bf0ff974ceabc0ad58e589b28797c684

[48] S. P. Priyadharshini and J. Ramkumar, "Mappings Of Plithogenic Cubic Sets," *Neutrosophic Sets and Systems*, vol. 79, pp. 669–685, 2025, doi: 10.5281/zenodo.14607210.

[49] S. P. Geetha, N. M. S. Sundari, J. Ramkumar, and R. Karthikeyan, "ENERGY EFFICIENT ROUTING IN QUANTUM FLYING AD HOC NETWORK (Q-FANET) USING MAMDANI FUZZY INFERENCE ENHANCED DIJKSTRA'S ALGORITHM (MFI-EDA)," *J Theor Appl Inf Technol*, vol. 102, no. 9, pp. 3708–3724, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85197297302&partnerID=40&md5=72d51668bee6239f09a59d2694df67d6

[50] R. Jaganathan, S. Mehta, and R. Krishan, "Preface," *Bio-Inspired Intelligence for Smart Decision-Making*, pp. xix–xx, 2024, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85195725049&partnerID=40&md5=7a2aa7adc005662eebc12ef82e3bd19f

[51] J. Ramkumar and R. Vadivel, "CSIP—cuckoo search inspired protocol for routing in cognitive radio ad hoc networks," in *Advances in Intelligent Systems and Computing*, 2017, pp. 145–153. doi: 10.1007/978-981-10-3874-7_14.

[52] B. Suchitra, J. Ramkumar, and R. Karthikeyan, "Frog Leap Inspired Optimization-Based Extreme Learning Machine For Accurate Classification Of Latent Autoimmune Diabetes In Adults (LADA)," *J Theor Appl Inf Technol*, vol. 103, no. 2, pp. 472–494, 2025, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

85217140979&partnerID=40&md5=9540433c16d5ff0f6c2de4b8c43a4812

[53] R. Jaganathan and V. Ramasamy, "Performance modeling of bio-inspired routing protocols in Cognitive Radio Ad Hoc Network to reduce end-to-end delay," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 1, pp. 221–231, 2019, doi: 10.22266/IJIES2019.0228.22.

[54] J. Ramkumar, R. Karthikeyan, and M. Lingaraj, "Optimizing IoT-Based Quantum Wireless Sensor Networks Using NM-TEEN Fusion of Energy Efficiency and Systematic Governance," in *Lecture Notes in Electrical Engineering*, 2025, pp. 141–153. doi: 10.1007/978-981-97-6710-6_12.

[55] B. Suchitra, J. Ramkumar, and R. Karthikeyan, "Frog Leap Inspired Optimization-Based Extreme Learning Machine For Accurate Classification Of Latent Autoimmune Diabetes In Adults (LADA)," *J Theor Appl Inf Technol*, vol. 103, no. 2, pp. 472–494, 2025, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85217140979&partnerID=40&md5=9540433c16d5ff0f6c2de4b8c43a4812

[56] R. Jaganathan, S. Mehta, and R. Krishan, *Bio-Inspired intelligence for smart decision-making*. 2024. doi: 10.4018/9798369352762.

[57] J. Ramkumar, V. Valarmathi, and R. Karthikeyan, "Optimizing Quality of Service and Energy Efficiency in Hazardous Drone Ad-Hoc Networks (DANET) Using Kingfisher Routing Protocol (KRP)," *International Journal of Engineering Trends and Technology*, vol. 73, no. 1, pp. 410–430, 2025, doi: 10.14445/22315381/IJETT-V73I1P135.

[58] J. Ramkumar, B. Varun, V. Valarmathi, D. R. Medhunhashini, and R. Karthikeyan, "Jaguar-Based Routing Protocol (Jrp) For Improved Reliability And Reduced Packet Loss In Drone Ad-Hoc Networks (DANET)," *J Theor Appl Inf Technol*, vol. 103, no. 2, pp. 696–713, 2025, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85217213044&partnerID=40&md5=e38a375e46cf43c95d6702a3585a7073

[59] J. Ramkumar, R. Karthikeyan, and V. Valarmathi, "Alpine Swift Routing Protocol (ASRP) for Strategic Adaptive Connectivity Enhancement and Boosted Quality of Service in Drone Ad Hoc Network (DANET)," *International Journal of Computer Networks and Applications*, vol. 11, no. 5, pp. 726–748, 2024, doi: 10.22247/ijcna/2024/45.

[60] J. Ramkumar, C. Kumuthini, B. Narasimhan, and S. Boopalan, "Energy Consumption Minimization in Cognitive Radio Mobile Ad-Hoc Networks using Enriched Ad-hoc On-demand Distance Vector Protocol," in *2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022*, 2022. doi: 10.1109/ICACTA54488.2022.9752899.

[61] R. Vadivel and J. Ramkumar, "QoS-enabled improved cuckoo search-inspired protocol (ICSIP) for IoT-based healthcare applications," in *Incorporating the Internet of Things in Healthcare Applications and Wearable Devices*, 2019, pp. 109–121. doi: 10.4018/978-1-7998-1090-2.ch006.

[62] J. Ramkumar, B. Varun, V. Valarmathi, D. R. Medhunhashini, and R. Karthikeyan, "Jaguar-Based Routing Protocol (Jrp) For Improved Reliability And Reduced Packet Loss In Drone Ad-Hoc Networks (DANET)," *J Theor Appl Inf Technol*, vol. 103, no. 2, pp. 696–713, 2025, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85217213044&partnerID=40&md5=e38a375e46cf43c95d6702a3585a7073