An Effective Routing Algorithm to Minimize the UAV Routing Time and Extend the Network Lifetime in Clustered IoT Network

Hanif ZAFOR [1], Tasher Ali SHEIKH [2], Nabajyoti MAZUMDAR [3], Amitava NAG [1]

[1] Dept. of Computer Science and Engineering, Central Institute of Technology Kokrajhar, Assam-783370, India
[2] National Institute of Electronics and Information Technology (NIELIT), Bhubaneswar, Odisha 751013, India
[3] Dept. of Information Technology, Indian Institute of Information Technology Allahabad, UP-211015, India

zafor90@gmail.com, {tasher.ece@gmail.com, nabajyoti@iiita.ac.in, amitava.nag@cit.ac.in}

Submitted November 27, 2024 / Accepted March 28, 2025 / Online first May 19, 2025

Abstract. *Recently, unmanned aerial vehicles (UAVs) have become more popular due to their ease of adaptability and capability to carry out a variety of activities, including the delivery of services, monitoring and surveillance in military and civilian contexts. One of the most significant challenges in UAV operation is ensuring maximum network lifetime and management of their limited battery life. To solve these problems, we have proposed an effective routing algorithm that finds the best route to minimize UAV routing time and extend network lifetime. This is performed using the Ant Colony Optimization with Local Search (ACO-LS) algorithm for data collection from the clustered IoT network by UAV to ensure maximum network lifetime. It solved the routing problem in the minimum time in the presence of multiple charging stations and optimized the routing path. The simulation was carried out using various performance metrics: network lifetime (NT), energy consumption (EC), number of alive nodes (NAN), and packet delivery percentage (PDP). These parameters were compared with some existing algorithms such as Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA) and found that our proposed algorithm performs better in terms of higher NT, less EC, more NAN, and higher PDP than the existing algorithms ACO, PSO, and GA.*

Keywords

Internet of Things (IoT), data collection (DC), unmanned aerial vehicles (UAVs), ant colony optimization (ACO), local search (LS), particle-swarm optimization (PSO), genetic algorithm (GA)

1. Introduction

The unmanned aerial vehicles (UAVs) are being widely used because of their adaptability and capacity to carry out a variety of activities in various fields such as agriculture, transportation for delivery of services, environmental monitoring, and surveillance in both military and civilian contexts [1–3]. The main significant challenges of UAV operations in internet of things (IoT) network are ensuring efficient routing path for data collection, management of limited battery life of the UAV [4–7]. In order to address these issues, multiple charging stations can be deployed in the operational area to enable UAV to recharge their batteries during their routing time. However, in the presence of multiple charging stations introduces new routing challenges, such as determining which charging station to use in the network and when, and also considering the UAV remaining battery life and mission requirements [8–9]. Therefore, the design of an effective routing algorithm that can account for the presence of multiple charging stations is essential for optimizing the use of UAV in various applications. The routing of UAV in the presence of numerous charging stations has gain importance in recent years [10–14]. The objective is to determine the best route for a UAV to collect data and travel to several charging stations to recharge its battery while taking into account a number of variables, including the charging stations locations, the UAVs battery level, and the energy used during routing [15], [16].

In article [17], a layered UAV swarm network architecture is developed to address the issues of low latency service needs and dynamic topology for efficient routing, and the ideal number of UAVs is examined. The low latency and data traffic flows that maximize the packet delivery ratio and enhance route stability based on the predicted connection stability of UAVs and minimize the delay. In paper [18], extend the lifetime of the WSN through energy harvesting and sensor energy consumption. The primary goal is to minimize the energy consumption by sensor node and optimize the hovering position and duration of the UAVs during data collection.

In [19], reducing the overall distance traveled by the UAV while analyzing congestion where there are many charging stations. This problem may be resolved by using a multi-head heterogeneous attention system in conjunction with

DOI: 10.13164/re.2025.0342

a deep reinforcement learning based approach to develop an automated route design strategy. In paper [20], minimize the average delivery time to user by using an innovative approach that uses the optimum deployment strategy. A suboptimal method is used while moving to the charging stations one after the other so that every relocation causes an average distance travelled. In paper [21], a novel MILP model is developed for a low cost design approach in order to handle the UAV route and charging station scheduling for belt conveyor monitoring. The MILP model depicts a difficult real world issue for determining the right number of UAVs for tracking the system, how to use charge stations and the optimal routes.

In article [22], two routing methods have been used: i) intra-cluster and ii) inter-cluster. In the inter-cluster routing the CH apply the ACO algorithm to identify the optimum path to the BS. This minimizes the algorithmic latency and offers a smoother operation by combining ACO with clustering. In [23], primary goal is to maximize the network lifetime by specifying link cost, remaining power, and necessary transmission energy utilization link cost. A new pheromone update operator was created to incorporate the use of energy and travels into the routing decision. In the article [24], a novel ACO-based mobile sink path selection method is proposed for WSN that reduces latency and maximizes network lifespan. A proficient method is employed to identify a nearly ideal combination of Rendezvous point (RP) and mobile sink travel path in order to accomplish the intended goal.

In our article, we have proposed an effective routing algorithm for data collection from the cluster heads (CHs) of IoT nodes using a UAV with charging stations in the network. The UAV acts as a mobile data collector that gathers information from the CHs throughout the network. To determine the optimal routing path, we employ an Ant Colony Optimization with Local Search (ACO-LS) algorithm. This approach creates a virtual ant colony that explores the search space and updates pheromone trails based on the quality of the solutions found. The algorithm takes into account the UAV battery level and the locations of charging stations, thereby identifying a path that minimizes travel distance while ensuring that data is collected from the clustered IoT nodes within the designated time window [25], [26]. To apply ACO-LS to the UAV routing problem with multiple charging stations, the problem is represented as a complete network with IoT nodes, charging stations, base station, and connecting path between these IoT nodes. The algorithm then selects the shortest path based on both the pheromone levels and the distances to the charging stations. By integrating dynamic programming into the ACO framework, the ACO-LS algorithm also determines the optimal charging strategy for the UAV, when and where to charge to maximize its range and minimize travel time. Overall, the ACO-LS algorithm is an effective optimization algorithm for solving the UAV routing problem in scenarios with multiple charging stations, as it provides an optimal solution that balances the trade-off between the distance traveled and the UAV charging time during data collection from the cluster heads of IoT nodes in the network.

The major contributions of our proposed article are summarized as follows: This article describes a clustered distributed network model for optimizing IoT networks. In this model, IoT nodes are randomly deployed and then grouped into different clusters. The cluster heads (CH) of the clusters are selected based on the residual energy of the IoT nodes, distance, and degree of connectivity between nodes. This clustering strategy significantly enhances energy efficiency, reduces communication overhead, and extends the network's lifetime. The network design presents an advanced ACO-LS based routing algorithm that guides UAV in collecting data efficiently and locating charging stations within the network. The hybrid ant colony optimization (ACO) with local search (LS) technique ensures UAV follow the most optimized routes, balancing energy consumption and reducing the time to travel across the network.

The main motive of our work is to minimize the UAV routing time and to enhance the lifetime of the network by ensuring that all data are collected throughout the network using ACO-LS, an effective routing algorithm. This approach results in an optimized routing path that maintains operational efficiency. The proposed algorithm ACO-LS is compared to existing algorithms like ACO, PSO, and GA with performance metrics- network lifetime, energy consumption, number of alive nodes, and packet delivery percentage. The proposed method demonstrate superior performance in multiple aspects and significantly improves the network's lifetime, consumes less energy, increases the number of active nodes, and ensures a higher packet delivery rate. These remarkable enhancements make the method not only more energy efficient but also more reliable and scalable, offering a robust solution for large-scale IoT applications that relay on UAV for data collection.

The paper is organized as follows: In Sec. 2, proposed network model is explained in detail. In Sec. 3, proposed algorithm of this work is presented. The simulation results are discussed in Sec. 4. Finally, Section 5 concludes the work and presents future works.

2. Proposed Network Model

The network model aims to determine the most efficient routing strategy for the UAV while allowing for the presence of multiple charging stations. The objective is to extend the network lifetime and maximize the number of completed tasks while minimizing the total path of UAV travel within a given time window. To achieve this, the UAV must visit all required locations, effectively manage its battery capacity, and strategically use the available charging stations along the optimal routes. The step by step strategy of our proposed work is presented in the flow chart diagram in Fig. 1.

We have considered a distributed network model based on clusters, where the clustered distribution implies that IoT nodes are deployed randomly and then grouped together in distinct clusters. For partitioning the complete network, we have used the K-Means clustering algorithm presented in Algorithm 1 [27]. The list of parameters used is presented in Tab. 1.

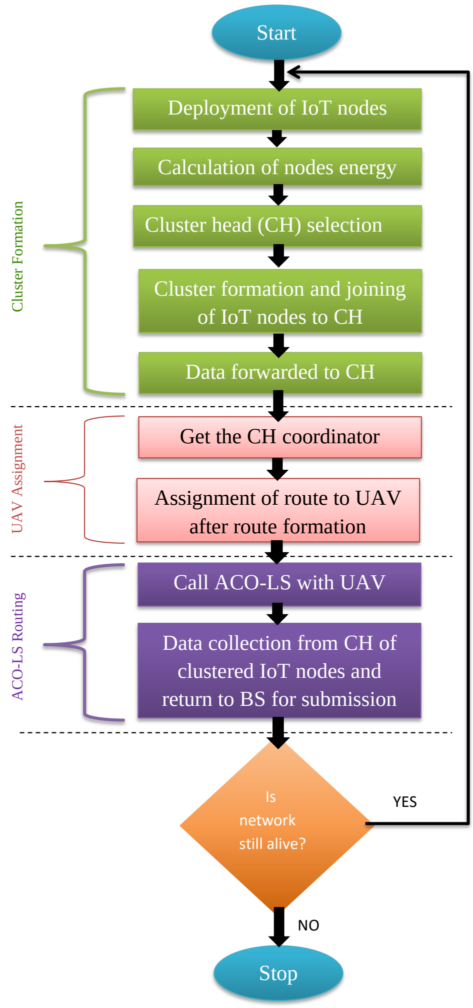

Fig. 1. Flow chart diagram of proposed model for UAV routing.

Algorithm 1. K-Means clustering algorithm.

1: **Input:** Dataset $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$, where each $x_i \in \mathbb{R}^d$; number of clusters K; initial cluster heads $\{CH_1, CH_2, \ldots, CH_K\}$ selected from the IoT nodes.

2: **Repeat until convergence:**

a. **Assignment step:** For each data point $x_i \in \mathcal{D}$, assign it to the nearest cluster:
$$S_j = \{x_i \in \mathcal{D} \mid \|x_i - CH_j\|^2 \le \|x_i - CH_k\|^2, \; \forall k \in \{1, 2, \ldots, K\}\}, \quad \text{for } j = 1, 2, \ldots, K.$$

b. **Update step:** For each cluster $j = 1, 2, \ldots, K$, update the cluster head as follows:
$$CH_j \leftarrow \frac{1}{|S_j|} \sum_{x_i \in S_j} x_i .$$

3: **Termination:** Stop when the cluster head change negligibly between iterations.

4: **Output:** The K clusters $\{S_1, S_2, \ldots, S_K\}$ with their corresponding cluster heads $\{CH_1, CH_2, \ldots, CH_K\}$.

The proposed network model for UAV routing to collect data from the cluster head (CH) of the clustered IoT nodes in the presence of multiple charging stations is shown in Fig. 2. We assumed total N number of IoT nodes, a UAV to collect data from the CH of the IoT nodes, the base station (BS) in the presence of C number of charging stations in the network. In clustered network model UAV will visit the CH for data collection by choosing the shortest route, for that it requires an efficient routing algorithm. When multiple requests come from IoT nodes with a time window for data collection, this algorithm tries to fulfill all requests within the time window. While the UAV travels to the IoT nodes, the battery of the UAV may be low, so it is required to recharge the battery of the UAV before traveling to the next CH. The UAV travels to the charging station for recharging, then it travels to the IoT nodes to collect data before the deadline.

In this article, we have explained how UAV should be routed in the network to collect data from the IoT nodes when there are several recharging stations available. The data or packets are delivered using a fleet of UAV with fixed flying ranges and uniform loading capacity. The UAV may need to stop at charging stations in order to resume their journey's since as they follow their planned itineraries, their batteries run out proportionally to the distance travelled. In the context of routing UAV with multiple charging stations, it is typically assumed that each UAV departs from BS or central depot with a fully charged battery and returns to the same depot after completing it's tasks.

Parameter	Definition of the parameter
n	Total number of data points
d	Number of features (dimensions) in each data point
K	Number of clusters
x_i	A data point in the dataset, where $i \in \{1, \ldots, n\}$
CH_i	The cluster heads of clusters i, where $i \in \{1, \ldots, K\}$
S_j	The set of data points assigned to cluster j
$\|x_i - CH_i\|^2$	Euclidean distance squared between a node and a cluster head

Tab. 1. Definition of parameters used in K-Means clustering algorithm.

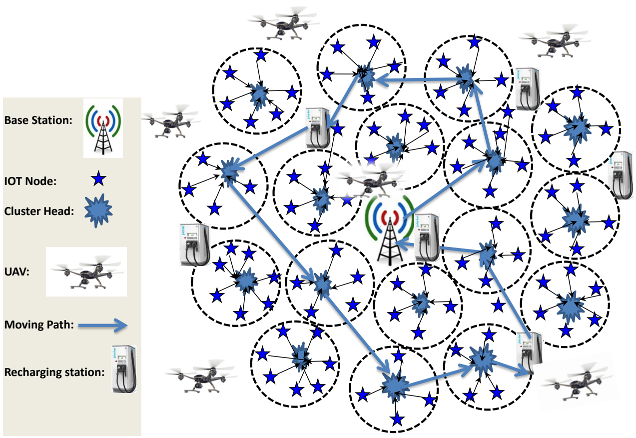

Fig. 2. Block diagram of proposed network model for routing of UAV.

We considered the complete network as $G = \langle U, V \rangle$, where G is the collection of IoT nodes and connecting paths, $U = 1, \ldots, N$ denotes the set of IoT nodes, and C represent the set of charging stations. Let $U^{'}$ be the set of IoT nodes with $U^{'} = U \cup C$. It appears that you are referring to a mathematical notation for a set of IoT nodes, where 0 and $N + 1$ represent the starting and ending depots i.e BS respectively. In this notation, U_0 is the set of nodes that includes the starting depot 0 but not the ending depot $(N+1)$, i.e $U_0^{'} = U^{'} \cup \{0\}$ and $U_{N+1}^{'} = U^{'} \cup \{N + 1\}$ is the set of nodes that includes both the starting depot 0 and the ending depot $(N + 1)$. To be more precise, U_0 can be defined as $U_0 = 1, 2, \ldots, N$ where N is the total number of nodes excluding the depots. It seems like you have provided a context for a problem on a complete network with $N + 1$ nodes, where distance between the nodes and travel time associated with it. In this network, there are $N + 1$nodes represented by numbers 1 to $N + 1$. Each path in the network is represented by an ordered pair (i, j), where i and j are distinct nodes in the network.

Where V is the set of connecting path between the nodes in the network, and each path is associated with a distance $D_{i,j}$ and a travel time $T_{i,j}$. The traveling path consumes a certain amount of the battery at a constant rate of r. A positive demand d_i, a service time s_i, and a temporal window $[t_i, t_j]$ for each path $i \in V$. A positive demand k_i, a service time T_i, and a time frame $[t_i, t_j]$ are present for each path $i \in V$, and the UAV has batteries with E capacity. The decision variables i, u_i, and v_i represent the IoT node i, service beginning time, and remaining charge level, respectively. If the path (i, j) is traversed, the binary choice variable Υ_{ij} takes 1, otherwise it takes 0. The notation used in the network model formulation with definition is shown in Tab. 2.

The mathematical representation of the proposed model is formulated as a mixed integer linear program as follows:

$$\min \prod = p_{\mathrm{f}} \sum_{j \in N'} \Upsilon_{0j} + p_{\mathrm{t}} \sum_{i,j \in N'} \Upsilon_{ij} T_{ij} + p_{\mathrm{r}} \sum_{i \in C} k_i \Theta_i + p_{\mathrm{w}} \sum_{i \in N} \alpha_i, \tag{1}$$

$$\sum_{j \in N_{n+1}^{'}, i \neq j} \Upsilon_{ij} = 1, \forall i \in N, \tag{2}$$

$$\sum_{j \in N_{n+1}^{'}, i \neq j} \Upsilon_{ij} \leq 1, \forall i \in C, \tag{3}$$

$$\sum_{i \in N_0^{'}, i \neq j} \Upsilon_{ij} = \sum_{i \in N_{n+1}^{'}, i \neq j} \Upsilon_{ji}, \forall j \in N^{'}. \tag{4}$$

The objective of Equation (1) is to minimize the distance traveled. Equation (2) ensures that each node in the network is connected to exactly one other, except for the facilities. Equation (3) limits the number of connections that can be made to facilities to at most one. Equation (4) ensures the flow is conserved at each node in the network.

Notation	Definition of the notations
a_i	Demand of IoT nodes i
b_i	Nodes service period i
h_i	Service at vertex's earliest possible time i
k_i	Recent beginning of service at vertex i
P_{f}	UAV's fixed cost
P_{t}	Cost of travel for each individual
P_{r}	Recharging cost for each unit
P_{w}	Waiting expense per unit
w_i	Time when the vertex's service began i
α_i	UAV's vertex based waiting time $i \in V$
β_i	UAV's remaining battery charge level node i
θ_i	In partial recharge, the amount of recharging
η_i	Remaining charge level after UAV departs i
x_i	UAV battery state after departing C station
α	Pheromone trail exponent
β	Heuristic information exponent
Ψ_{best}	Current best solution
ζ_{best}	Best feasible solution of ACO-LS algorithm
γ_{Curr}	Current solution found after of LS algorithm
γ_{best}	Best solution found after of LS algorithm

Tab. 2. Notations used in the network model formulation.

$$\omega_i + (T_{ij} + b_i)\Upsilon_{ij} - K_0(1 - \Upsilon_{ij}) \leq w_j, \quad \forall i \in N_0^{'}, \forall j \in N_{n+1}^{'}, i \neq j, \tag{5}$$

$$\omega_i + T_{ij}\Upsilon_{ij} + Ql_i k_i - (k_0 + Ql)(1 - \Upsilon_{ij}) \leq w_j, \quad \forall i \in C, \forall j \in N_{n+1}^{'}, i \neq j, \tag{6}$$

$$h_j \leq w_j \leq K_j, \forall j \in N_{n+1}^{'}, \tag{7}$$

$$0 \leq \beta_j \leq \beta_i - a_i \Upsilon_{ij} + L(1 - \Upsilon_{ij}), \quad \forall i \in N_0^{'}, \forall j \in N_{n+1}^{'}, i \neq j. \tag{8}$$

$$0 \leq \beta_0 \leq L \tag{9}$$

Equation (5) ensures that the transit time from node i to j plus any setup time at i does not exceed the latest allowable arrival time at node j. Equation (6) ensures that the transit time from facility i to j plus any production time in facility i does not exceed the latest allowable arrival time at node j, while also taking into account the capacity of facility i. Equation (7) enforces the time window constraint in each of the networks. Equations (8) and (9) guarantee demand fulfillment at all customers by assuring a nonnegative cargo load upon arrival at any node including the BS.

$$0 \leq \eta_j \leq \eta_i - (\rho.D_{ij})\Upsilon_{ij} + l(1 - \Upsilon_{ij}), \quad \forall i \in N, \forall j \in N_{n+1}^{'}, i \neq j, \tag{10}$$

$$0 \leq \eta_j \leq x_i - (\rho.D_{ij})\Upsilon_{ij} + l(1 - \Upsilon_{ij}), \quad \forall i \in C, \forall j \in N_{n+1}^{'}, i \neq j, \tag{11}$$

$$x_i = k_i(\eta_i + \theta_i) + \mu_i l, \forall i \in C, \tag{12}$$

$$x_i \leq l, \forall i \in C, \tag{13}$$

$$\Upsilon_{ij} \in \{0, 1\}, \forall i \in N_0^{'}, \forall j \in N_{n+1}^{'}, i \neq j, \tag{14}$$

$$\mu_j, k_j \in \{0, 1\}, \forall j \in C. \tag{15}$$

Equations (10) and (11) ensure that the battery charge never falls below 0. Equation (12) ensures that the battery of UAV at charging station is equal to partial recharging or full recharging done. Equation (13) ensures that the battery state of UAV after charging station departs is less or equal to the actual battery power of UAV. Equation (14) is the binary variable equal to 1 if the path is traversed otherwise 0, meaning that no recharging done. Equation (15) whether charging done at the station or not if it is done then it's value = 1, otherwise 0.

3. Proposed Algorithm

The proposed algorithm section is represented with solution strategy of our article is explained below.

3.1 Ant Colony Optimization-Local Search (ACO-LS)

An updated ACO with LS algorithm is suggested to address the UAV routing problem with the presence of multiple charging stations. The main procedure for improving ACO with LS is explained and the pseudo-code is presented in Algorithm 2.

Algorithm 2. Ant Colony Optimization-Local Search.

Initialization: Set pheromone trail $\tau \leftarrow 1$, best quality of the solution $\zeta_{\text{best}} \leftarrow 0$, initial solution $\Psi_0 \leftarrow 0$, and evaporation factor $\rho \leftarrow 0$.
for $i \leftarrow 1$ **to** $\max_{\text{iteration}}$ **do**
  for $k \leftarrow 1$ **to** N **do**
    Construct an initial solution ϕ_0 using a constructive heuristic.
    for $j \leftarrow 1$ **to** NR **do**
      Construct a new solution Ψ_0 using an insertion heuristic.
      Improve Ψ_0 by applying Algorithm 3.
      Update the pheromone trail τ based on the quality of the current solution.
    end for
    if $\Psi_{\text{best}} > \zeta_{\text{best}}$ **then**
      Update $\zeta_{\text{best}} \leftarrow \Psi_{\text{best}}$.
    end if
  end for
  Evaporate the pheromone trail: $\tau \leftarrow \tau \cdot (1 - \rho)$.
  Reinforce the pheromone trail with the best solution: $\tau \leftarrow \zeta_{\text{best}}$.
end for
Output: ζ_{best}.

3.2 Construction of the First Colony

The usage of pheromone trails in an ant colonies is an effective illustration of how positive feedback mechanisms can result in the creation of complex behaviour and effective resource distribution in a decentralized system. The goal is to reduce the overall distance travelled while still allowing the UAV to effectively cover the full region. However, because the Nearest Recharging (NR) system does not take into consideration the battery constraint or the accessibility of recharging stations, it is not a workable solution for the original UAV problem. The classic probabilistic algorithm employed in ACO for choosing the IoT nodes has been improved by the distances and time windows (DTW) probabilistic model. It produces more effective and workable solutions since it considers both the time window and distance restrictions. The following is a presentation of the DTW probability model for ant k at the current location of IoT node i [28]:

$$F_{i,j}^{k} = \begin{cases} \dfrac{\lambda_{i,j}^{\theta} \cdot \gamma_{i,j}^{\phi} \cdot \delta_{i,j}^{\partial}}{\sum_{\varphi \in h_k(i)} \lambda_{i\varphi}^{\theta} \cdot \gamma_{i\varphi}^{\phi} \cdot \delta_{i\varphi}^{\partial}}, & \text{if } h_k \in h_k(i) \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

where $F_{i,j}^{k}$ determines the likelihood of combining IoT nodes i and j along the path of ant k, $h_k(i)$ is the collection of IoT nodes available for selection. The λ_{ij} , γ_{ij} and δ_{ij} indicates the density of pheromones, distance horizon visibility and path i to j time windows respectively. The relative weights of the pheromone trails and the two visibility levels are indicated by θ, ϕ and ∂. The expectation factors in this study γ_{ij} and δ_{ij} are established as follows:

$$\gamma_{ij} = \frac{1}{D_{ij}}, \tag{17}$$

$$\delta_{ij} = \frac{1}{t_j} \tag{18}$$

where t_j is the most recent arrival time of the time windows for the IoT node j, and D_{ij} is the length of edge from node i to j. The time window restriction δ_{ij} is represented as a penalty function that punishes nodes that violate the time window limitations. It accounts for the length of the time frame, the arrival time, and the waiting time. The penalty is zero if the waiting period is less than or equal to 0. Otherwise, the penalty is based on how long you have to wait. The probabilistic DTW model allows the ants in the colony to select the subsequent IoT based on the pheromone concentration, visibility, and time window constraints. This leads to solutions that are practical and satisfy the limited time window in UAV routing with numerous charging stations.

3.3 Insertion Heuristic

The insertion heuristic is a typical method for solving optimization problems, especially when it comes to UAV routing issues, including the refueling station or charging station. The insertion heuristic works by incrementally adding a charging station to a route until it is practical. Finding the path that requires the greatest charge to become practicable is the first step in the procedure, and then the charging station that incurs the least additional expense is added to the route. Once all routes are feasible, the heuristic repeats the process with the remaining impossible routes. In general, using the insertion heuristic to solve UAV routing issues that require charging or refueling stations can be successful, and the effectiveness of the algorithm can vary depending on the particular problem and the caliber of the initial solution.

3.4 Local Search (LS) Algorithm

The local search algorithm is shown in Algorithm 3. The local search technique used in this study combines removal and insertion operators in an effort to increase the quality of the solutions. The removed IoT node is then fixed by reinserting it using insertion operators into the solution. Its crucial to keep in mind during the local search process that the placement and number of recharging stations may change depending on where and how many IoT nodes are removed or added. Therefore, only non-recharging routes without stations are subject to the removal and insertion operators. The insertion heuristic is used to create new workable solutions once the removal and insertion operators have been applied. The time violated IoT node is removed from the route and added to a set of $S^{\text{unvisited}}$ IoT nodes if the final solution breaks any time windows. In the following iterations of the local search, these unexplored IoT nodes are taken into account. If the answer is workable, it is included in the list of workable answers X_0. The local search is carried out repeatedly until the maximum number of iterations is reached or a feasible solution is found to be a more alive node in the network.

Algorithm 3. Local Search.

Initialization: $\gamma_{\text{curr}} \leftarrow 0$, $\gamma_{\text{best}} \leftarrow 0$, $\psi_{\text{removal\_list}} \leftarrow 0$, and $i \leftarrow 1$.
  Generate an initial feasible solution γ_{curr}.
  Set the best solution $\gamma_{\text{best}} \leftarrow \gamma_{\text{curr}}$.
for $i = 1$ **to** $\max_{\text{iteration}}$ **do**
  Randomly select a removal operator and form a list ψ_{removal}.
  Remove the nodes in ψ_{removal} from γ_{curr}.
  Reinsert the removed nodes in γ_{curr} using a random insertion operator.
  Improve γ_{curr} by applying an insertion heuristic.
  if $\text{Cost}(\gamma_{\text{curr}}) < \text{Cost}(\gamma_{\text{best}})$ **then**
    Update $\gamma_{\text{best}} \leftarrow \gamma_{\text{curr}}$.
  end if
  Generate a new solution $\gamma_{\text{curr}}^{\text{new}}$ by removing all charging stations from the current routes in γ_{curr}.
  Set $\gamma_{\text{curr}} \leftarrow \gamma_{\text{curr}}^{\text{new}}$.
end for
Output: γ_{best}.

Algorithms [ref. no.]	Time complexity
GA [31]	$O(M \times N^2)$
PSO [30]	$O(M \times N^2)$
ACO [29]	$O(N^2)$
ACO-LS [Proposed]	$O(N^2)$

Tab. 3. Time complexity comparison with existing algorithms.

3.5 Time Complexity Analysis

The time complexity (TC) of our proposed ant colony optimization-local search (ACO-LS) algorithm is a combination of the ACO and the LS algorithm. The TC of the ACO algorithm is $O(I \times k \times N^2)$, where I is the number of iterations, k is the number of ants and N is the number of IoT nodes including cluster heads and recharging stations. The local search improves solutions by examining neighbouring options. The TC of the local search is $O(I \times N^2)$. The $O(N^2)$ is for evaluating neighbours solution. The total TC of our proposed local search algorithm for the optimization of the ant colony (ACO-LS) is $O(I \times k \times N^2)$+ $O(I \times N^2)$ = $O(I \times k \times N^2)$, now neglecting the values of I and k. The final time complexity of the ACO-LS algorithm $O(N^2)$.

The TC of the existing algorithms are: The TC of the ACO [29] algorithm is $O(N^2)$, the TC of the PSO [30] algorithm is $O(M \times N^2)$ and the TC of the GA [31] algorithm is $O(M \times N^2)$, where N is the number of sensor nodes and M is the number of cluster heads. The comparison of TC of our proposed algorithm with the existing algorithms is shown in Tab. 3.

4. Simulation Results

The proposed approach is designed to increase the lifetime of the network using efficient routing and data collection using UAV. There are many definitions of the lifetime of the network in the WSN literature [32]. We have defined the network lifetime based on the time at which the first node dies and the last node dies in the network. To prove the efficiency of proposed approach, we have applied the ACO-LS based routing algorithm for data collection from IoT nodes by UAV with multiple charging stations. We have compared the network lifetime based on the first node dies and the last node dies in the network and number of alive nodes with the number of iterations, the total energy consumption and the packet delivery percentage with number of nodes. Furthermore, this proposed method is compared to existing methods such as ACO, PSO, and GA [29–31].

The simulations were carried out on a computer system with an AMD Ryzen-7 processor running at a speed of 1.8 GHz and 16 GB of RAM using the Python 3.10 programming language. An AMD Ryzen-7 processor, which provides an excellent mix between processing capability and energy efficiency, is installed on the computer system utilized for the studies. The processor operating frequency, which affects the algorithm overall execution performance. The 16 GB RAM size ensures that there is enough memory to satisfy the experiment computing needs. Overall, the decision to use Python as the programming language and the use of an AMD Ryzen-7 processor running at a speed of 1.8 GHz and 16 GB of RAM give a strong basis for carrying out the computational experiments and generating trustworthy results to assess the effectiveness of the proposed ACO-LS algorithm.

Parameters	Value
Area of Network	$200 \times 200\,\mathrm{m}^2$
Number of UAV	1
Velocity of UAV	2 m/s
Number of IoT nodes	100 to 500
Number of iterations	0 to 3000
Types of IoT nodes	Static
Deployment IoT nodes	Random
Clustering method	K-Means
Initial energy of IoT nodes	0.5 J
E_{sensing}	0.001 J/task
$E_{\mathrm{processing}}$	0.005 J/task
$E_{\mathrm{UAV-tx}}$	0.0003 J/bit
E_{tx}	4.602 µJ/bit
E_{rx}	2.34 µJ/bit
E_{elec}	50 nJ/bit
E_{cpu}	7 nJ/bit

Tab. 4. Simulation parameter details.

We opted for a single UAV to minimize cost, as UAVs are more expensive than standard IoT nodes. To obtain a reliable average, we set the simulation to 3000 iterations. For simplicity and precision, IoT nodes are treated as static. Their deployment is random due to the challenging environment, making manual setup difficult. Our primary objective is to minimize energy consumption, so we assign low initial energy levels and optimize other energy parameters for the IoT nodes. The parameters used in the simulations for this paper are shown in Tab. 4. The value of the simulation parameters are based on values provided in articles [10], [33].

The simulation assumes that the base station is stationary, located in the center, and has unlimited energy. All IoT nodes in homogeneous environments are uniform, with equal and limited initial energy. They have a random uniform distribution and are deployed in a fixed position with a unique identity. The velocity of the UAV is constant. The UAV will collect data from the CHs. This device comes with a rechargeable battery and long-range transceiver.

Figure 3 shows the network lifetime vs. number of IoT nodes at which the first nodes die, and Figure 4 shows the network lifetime vs. number of IoT nodes at which the last node dies. The network lifetime increases with the increase in number of nodes in both the cases as shown in Figs. 3 and 4 by varying the number of nodes from 100 to 500.

The network lifetime percentage increased analysis compared to existing methods in terms of iterations at which the first nodes die: When IoT nodes = 100, network lifetime percentage increased by 8.62%, 28.57%, and 34.04%. When nodes = 200, network lifetime percentage increased by 3.24%, 8.69%, and 25%. When the IoT nodes = 300, network lifetime percentage increased by 5.12%, 8.46%, and 12.63%. When nodes = 400, network lifetime percentage increased by 2.32%, 10%, and 16.48% and when the nodes = 500, network lifetime percentage increased by 3.63%, 8.57%, and 16.92% compared to ACO, PSO, and GA. The average percentage of packet delivery increased by 4.58%, 12.85%, and 21.01% compared to ACO, PSO, and GA for IoT nodes = 100 to 500.

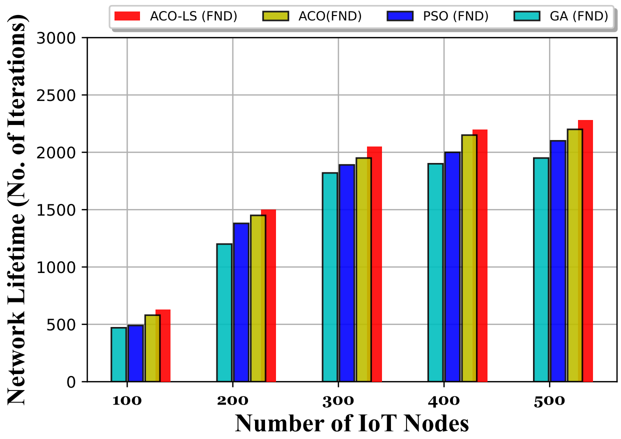

Fig. 3. Network lifetime based on first IoT node death in the network.

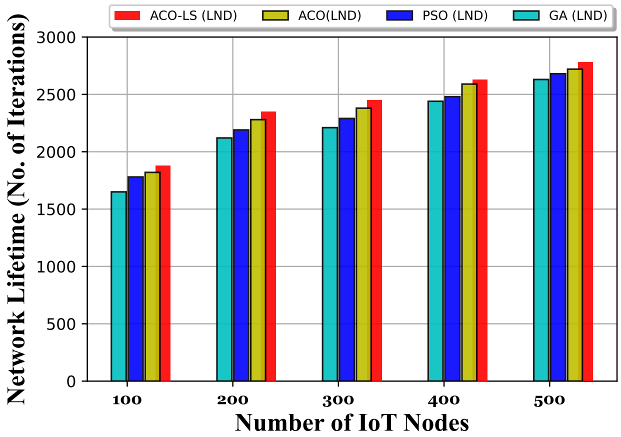

Fig. 4. Network lifetime based on last IoT node death in the network.

The network lifetime percentage increased analysis compared to existing methods in terms of iterations at which the last nodes die: When IoT nodes = 100, network lifetime percentage increased by 3.29%, 5.61%, and 13.93%. When nodes = 200, network lifetime percentage increased by 3.07%, 7.30%, and 10.84%. When the IoT nodes = 300, network lifetime percentage increased by 2.94%, 6.98%, and 10.85%. When nodes = 400, network lifetime percentage increased by 1.54%, 6.04%, and 7.78% and when the nodes = 500, network lifetime percentage increased by 2.20%, 3.73%, and 5.70% percentage compared to ACO, PSO and GA. The average percentage of packet delivery increased by 2.60%, 5.93%, and 9.82% compared to ACO, PSO and GA for IoT nodes = 100 to 500. It is clearly seen from Figs. 3 and 4 that our proposed method outperforms existing methods. Therefore, the lifetime of the network is longer compared to the other existing methods.

Figure 5 shows the number of alive nodes vs. number of iterations. The number of alive nodes decreases with an increase in the number of iterations. In Fig. 5 it is clearly observed that our proposed method has more number of alive nodes in each iteration ranges from 0 to 3000. The nodes start draining their energy completely between 400 to 2800 iterations. When iterations = 500, the number of alive nodes increased by 7.14%, 13.92%, and 23.28%. When iterations = 1000, number alive nodes increased by 19.04%, 38.88%, and 66.67%. When iterations = 1500, number alive nodes increased by 33.33%, 60%, and 100%. When iterations = 2000, number alive nodes increased by 37.5%, 120%, and 450%. When iterations = 2500, number alive nodes increased by 150%, 400%, and 490% compared to ACO, PSO, and GA. The proposed method maintains a more balanced number of alive nodes than the existing methods. This also achieves a longer network lifetime.

Figure 6 shows the total energy consumption vs. number of IoT nodes. The total energy consumption increases with an increase in the number of IoT nodes ranging from 50 to 1000. From Fig. 6, we can see that our proposed method has less energy consumption than the existing methods in each range of number of IoT nodes from 50 to 1000. When IoT nodes = 200, energy consumption is reduced by 4.27%, 23.80%, and 32.93%. When nodes = 400, energy consumption reduced by 12.93%, 43.98%, and 56.22%. When nodes = 600, energy consumption is reduced by 10.75%, 25.44%, and 29.66%.

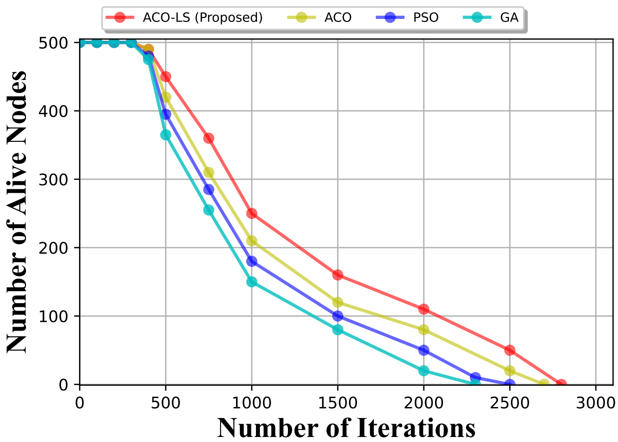

Fig. 5. Number of alive based on number of iterations in the network.

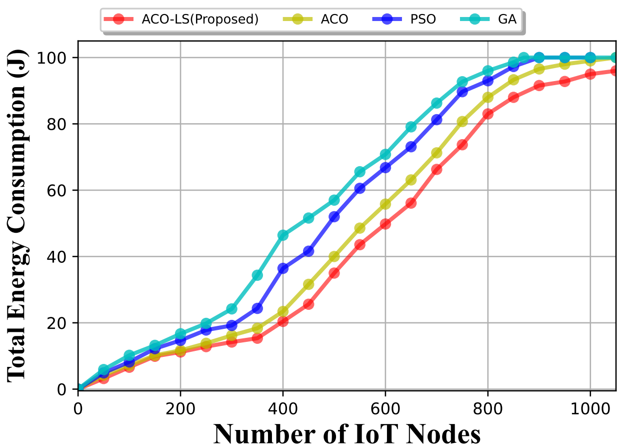

Fig. 6. Total energy consumption based on number of IoT nodes.

When nodes = 800, energy consumption reduced by 5.68%, 10.75%, and 13.54%. When nodes = 1000, energy consumption is reduced by 4.04%, 5%, and 5% compared to ACO, PSO, and GA. The average percentage of energy consumption is reduced by 7.53%, 21.79%, and 27.47% compared to ACO, PSO, and GA. From the graph and percentage analysis, we can say that our proposed method maintains low energy consumption that ensured the maximization of the lifetime of the network.

Figure 7 shows the percentage of packet delivery vs. number of IoT nodes at which the first IoT nodes die. The packet delivery percentage analysis at which the first nodes dies. When the nodes = 100, packet delivery percentage increased by 4.21%, 7.60%, and 8.79% . When nodes = 200, packet delivery percentage increased by 4.25%, 7.69%, and 10.11%. When nodes = 300, packet delivery percentage increased by 3.26%, 5.55%, and 6.74%. When nodes = 400, packet delivery percentage increased by 2.22%, 4.54%, and 4.54% and when the nodes = 500, packet delivery percentage increased by 2.27%, 5.88%, and 7.14% percentage compared to ACO, PSO, and GA. The average percentage of packet delivery increased by 3.24%, 6.25%, and 8.71% compared to ACO, PSO, and GA for IoT nodes range from 100 to 500. Through comparison from Fig. 7, we can observe that our proposed method has a higher packet delivery percentage than the existing methods in each range of number of IoT nodes from 100 to 500, which gives a smaller number of data loss in the network.

Figure 8 shows the packet delivery percentage vs. number of IoT nodes at which the last IoT node dies. The packet delivery percentage analysis at which the last nodes dies. When IoT nodes = 100, packet delivery percentage increased by 3.26%, 4.39%, and 7.95%. When nodes = 200, packet delivery percentage increased by 3.33%, 5.68%, and 8.13%. When the IoT nodes = 300, packet delivery percentage increased by 3.48%, 4.70%, and 5.95%. When nodes = 400, packet delivery percentage increased by 2.35%, 3.57%, and 4.81% and when the nodes = 500, packet delivery percentage increased by 3.65%, 4.93%, and 6.25% percentage

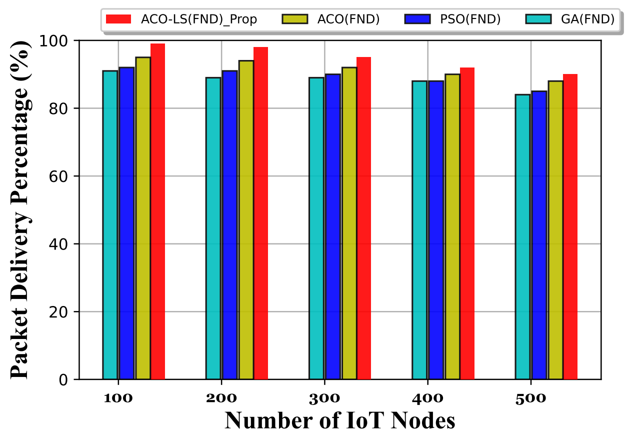

Fig. 7. Packet delivery percentage based on first IoT node death in the network.

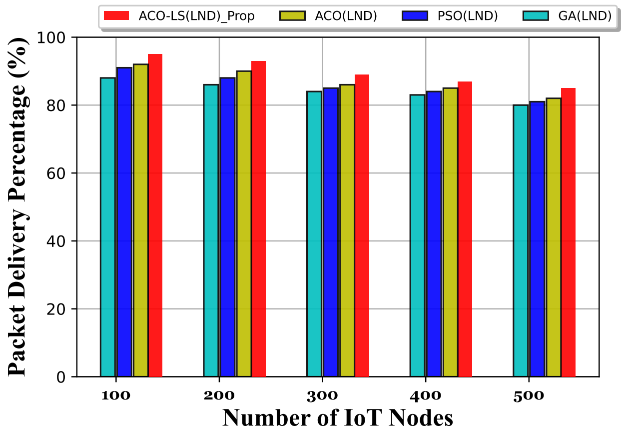

Fig. 8. Packet delivery percentage based on last IoT node death in the network.

compared to ACO, PSO, and GA. The average percentage of packet delivery increased by 2.48%, 4.65%, and 6.61% compared to ACO, PSO and GA for nodes = 100 to 500. Through comparison from Fig. 8 and percentage analysis, we can observe that our proposed method has a higher packet delivery percentage than the existing methods in each rang of number of IoT nodes from 100 to 500, which gives a smaller number of data loss in the network.

The justifications for the better efficiency of our proposed ACO-LS over existing algorithms [29–31] are listed below:

1. In ACO-LS, the cluster heads in the cluster are chosen on the basis of nodes with higher energy, degree of connectivity, and distance from the base station. Because IoT nodes for CH with these may withstand more load from normal IoT nodes. This clustering approach dramatically improves energy efficiency, decreases communication overhead, and increases network lifetime.

2. During cluster formation, our proposed ACO-LS uses a multi-objective optimization approach to choose its CH-based on factors such as residual energy of the CH, node to CH distance, and distance from CH to BS. However, in existing algorithms, non-CH nodes join a CH within their communication range based solely on distance. Such single-objective selection produces non-uniform burden distribution.

3. In terms of total energy consumption, our proposed ACO-LS consumed less energy compared to existing algorithms, as shown in Fig. 6 with detailed analysis. This method improves energy conservation and extends the lifetime of the network.

4. In ACO-LS, it performs better in terms of higher network lifetime, more number of alive nodes, and higher packet delivery percentage than the existing algorithms, as shown in Figs. 3–5, 7, and 8 with detailed analysis.

5. Conclusion

In this paper, we have proposed an Ant Colony Optimization with Local Search (ACO-LS) algorithm to solve the UAV routing problem, which improves the data collection efficiency and network lifetime in the IoT network. This algorithm optimizes UAV routing paths by forming a virtual ant colony that explores routes and updates pheromone trails based on pheromone density, minimizing travel distance and ensuring timely data collection from clustered IoT nodes. This algorithm improves routing efficiency by collecting data with multiple charging stations, resulting in greater network lifetime, lower energy consumption, more alive nodes, and higher packet delivery compared to traditional methods. We benchmark the proposed ACO-LS approach against Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and General Ant Colony Optimization (ACO), demonstrating superior performance across metrics.

In future work, UAV may be used to test the algorithm's performance in a network where the movement of mobile sinks, installed sensor nodes, and network size changes will be considered.

References

[1] MOHSAN, S. A. H., OTHMAN, N. Q. H., LI, Y., et al. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. *Intelligent Service Robotics*, 2023, vol. 16, no. 1, p. 109–137. DOI: 10.1007/s11370-022-00452-4

[2] FAN, B., LI, Y., ZHANG, R., et al. Review on the technological development and application of UAV systems. *Chinese Journal of Electronics*, 2020, vol. 29, no. 2, p. 199–207. DOI: 10.1049/cje.2019.12.006

[3] ZAFOR, H., SHEIKH, T. A., MAZUMDER, N., et al. Data collection and recharging of sensor node by mobile sink in wireless sensor network. *International Journal of Sensors, Wireless Communications and Control*, 2024, [In Press]. DOI: 10.2174/0122103279324801240826131750

[4] BORAH, J., SHEIKH, T. A., BORA, J. Dynamic cell sleeping mechanism: An energy-efficient approach for mobile 5G HetCN. *International Journal of Communication Systems*, 2023, vol. 36, no. 5, p. 1–14. DOI: 10.1002/dac.5422

[5] MENG, K., WU, Q., XU, J., et al. UAV-enabled integrated sensing and communication: Opportunities and challenges. *IEEE Wireless Communications*, 2024, vol. 31, no. 2, p. 97–104. DOI: 10.1109/MWC.131.2200442

[6] MU, J., ZHANG, R., CUI, Y., et al. UAV meets integrated sensing and communication: Challenges and future directions. *IEEE Communications Magazine*, 2023, vol. 61, no. 5, p. 62–67. DOI: 10.1109/MCOM.008.2200510

[7] ZAFOR, H., MAZUMDAR, N., NAG, A. A comparative study of survey papers based on energy efficient, coverage-aware, and fault tolerant in static sink node of WSN. In *Proceedings of the IEEE 9th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. Uttar Pradesh (India), 2022, p. 1–6. DOI: 10.1109/UPCON56432.2022.9986459

[8] CELIK, A., USTAOMER, E., SATOGLU, S. Single-drone energy efficient coverage path planning with multiple charging stations for surveillance. *An International Journal of Optimization and Control: Theories and Applications*, 2023, vol. 13, no. 2, p. 1–10. DOI: 10.11121/ijocta.2023.1332

[9] BACANLI, S. S., ELGELDAWI, E., TURGUT, B., et al. UAV charging station placement in opportunistic networks. *Drones*, 2023, vol. 6, no. 10, p. 1–21. DOI: 10.3390/drones6100293

[10] JLASSI, W., HADDAD, R., BOUALLEGUE, R., et al. Increase of the lifetime of wireless sensor network using clustering algorithm and optimal path selection method. *Radioengineering*, 2022, vol. 31, no. 3, p. 301–311. DOI: 10.13164/re.2022.0301

[11] SANTIN, R., ASSIS, L., VIVAS, A., et al. Matheuristics for multi-UAV routing and recharge station location for complete area coverage. *Sensors*, 2021, vol. 21, no. 5, p. 1–34. DOI: 10.3390/s21051705

[12] LIN, N., BAI, L., HAWBANI, A., et al. Deep reinforcement learning-based computation offloading for servicing dynamic demand in multi-UAV-assisted IoT network. *IEEE Internet of Things Journal*, 2024, vol. 11, no. 10, p. 17249–17263. DOI: 10.1109/JIOT.2024.3356725

[13] LIU, C. H., PIAO, C., TANG, J. Energy-efficient UAV crowd sensing with multiple charging stations by deep learning. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*. Toronto (Canada), 2020, p. 199–208. DOI: 10.1109/INFOCOM41043.2020.9155535

[14] RIBEIRO, R. G., COTA, L. P., EUZEBIO, T. A., et al. Unmanned-aerial-vehicle routing problem with mobile charging stations for assisting search and rescue missions in post disaster scenarios. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021, vol. 52, no. 11, p. 6682–6696. DOI: 10.1109/TSMC.2021.3088776

[15] LYU, T., AN, J., LI, M., et al. UAV-assisted wireless charging and data processing of power IoT devices. *Computing*, 2024, vol. 106, no. 3, p. 1–31. DOI: 10.1007/s00607-023-01245-y

[16] OJHA, T., RAPTIS, T. P., PASSARELLA, A., et al. Wireless power transfer with unmanned aerial vehicles: state of the art and open challenges. *Pervasive and Mobile Computing*, 2023, vol. 93, p. 1–34. DOI: 10.1016/j.pmcj.2023.101820

[17] KIM, S., KWAK, J. H., OH, B., et al. An optimal routing algorithm for unmanned aerial vehicles. *Sensors*, 2021, vol. 21, no. 4, p. 1–15. DOI: 10.3390/s21041219

[18] BAEK, J., HAN, S. I., HAN, Y. Optimal UAV route in wireless charging sensor networks. *IEEE Internet of Things Journal*, 2019, vol. 7, no. 2, p. 1327–1335. DOI: 10.1109/JIOT.2019.2954530

[19] FAN, M., WU, Y., LIAO, T., et al. Deep reinforcement learning for UAV routing in the presence of multiple charging stations. *IEEE Transactions on Vehicular Technology*, 2023, vol. 72, no. 5, p. 5732–5746. DOI: 10.1109/TVT.2022.3232607

[20] HUANG, H., SAVKIN, A. V. Deployment of charging stations for drone delivery assisted by public transportation vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 2021, vol. 23, no. 9, p. 15043–15054. DOI: 10.1109/TITS.2021.3136218

[21] RIBEIRO, R. G., JUNIOR, J. R., COTA, L. P., et al. Unmanned aerial vehicle location routing problem with charging stations for belt conveyor inspection system in the mining industry. *IEEE Transactions on Intelligent Transportation Systems*, 2019, vol. 21, no. 10, p. 4186–4195. DOI: 10.1109/TITS.2019.2939094

[22] KUMAR, P., AMGOTH, T., ANNAVARAPU, C. S. R. ACO-based mobile sink path determination for wireless sensor networks under non-uniform data constraints. *Applied Soft Computing*, 2018, vol. 69, p. 528–540. DOI: 10.1016/j.asoc.2018.05.008

[23] MOHAJERANI, A., GHARAVIAN, D. An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks. *Wireless Networks*, 2016, vol. 22, p. 2637–2647. DOI: 10.1007/s11276-015-1061-6

[24] SALEHPOUR, A. A., MIRMOBIN, B., AFZALI-KUSHA, A., et al. An energy efficient routing protocol for cluster-based wireless sensor networks using ant colony optimization. In *Proceedings of the International Conference on Innovations in Information Technology (IIT)*. Al Ain (UAE). 2008, p. 455–459. DOI: 10.1109/INNOVATIONS.2008.4781748

[25] DU, P., SHI, Y., CAO, H., et al. AI-enabled trajectory optimization of logistics UAVs with wind impacts in smart cities. *IEEE Transactions on Consumer Electronics*, 2024, vol. 70, no. 1, p. 3885-3897. DOI: 10.1109/TCE.2024.3355061

[26] LU, F., CHEN, N., LING, B. A deep reinforcement learning approach for solving the pickup and delivery problem with drones and time windows. *SSRN*, 2024, p. 1–23. DOI: 10.2139/ssrn.4684209

[27] SASIKUMAR, P., KHARA, S. K-means clustering in wireless sensor networks. In *Proceedings of the Fourth International Conference on Computational Intelligence and Communication Networks (CICN)*. Mathura (India), 2012, p. 140–144. DOI: 10.1109/CICN.2012.136

[28] ZHOU, W., GANG, W. An enhanced ACO-based mobile sink path determination for data gathering in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2022, p. 1–13. DOI: 10.1186/s13638-022-02145-z

[29] MAZUMDAR, N., ROY, S., NAG, A., et al. A buffer-aware dynamic UAV trajectory design for data collection in resource-constrained IoT frameworks. *Computers and Electrical Engineering*, 2022, vol. 100, no. C, p. 1–13. DOI: 10.1016/j.compeleceng.2022.107934

[30] LIANHAI, L., WANG, Z., TIAN, L., et al. A PSO-based energy-efficient data collection optimization algorithm for UAV mission planning. *PLOS One*, 2024, vol. 19, no. 1, p. 1–24. DOI: 10.1371/journal.pone.0297066

[31] BENMAD, I., DRIOUCH, E., KARDOUCHI, M. Data collection in UAV-assisted wireless sensor networks powered by harvested energy. In *Proceedings of the IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Helsinki (Finland), 2021, p. 1351–1356. DOI: 10.1109/PIMRC50174.2021.9569295

[32] DIETRICH, I., DRESSLER, F. On the lifetime of wireless sensor networks. *ACM Transactions on Sensor Networks*, 2009, vol. 5, no. 1, p. 1–39. DOI: 10.1145/1464420.1464425

[33] KRISHNA, M., YUN, S., JUNG, Y. M. Enhanced clustering and ACO-based multiple mobile sinks for efficiency improvement of wireless sensor networks. *Computer Networks*, 2019, vol. 160, p. 33–40. DOI: 10.1016/j.comnet.2019.05.019

About the Authors . . .

Hanif ZAFOR received B.E degree in Computer Science and Engineering from Jorhat Engineering College (JEC) in 2014. He completed his M.Tech in Computer Science and Engineering from the North Eastern Regional Institute of Science and Technology (NERIST) in 2016. He is currently pursuing Ph.D. in Computer Science and Engineering from Central Institute of Technology Kokrajhar (CITK), BTC, Assam, India. He has research publications in various international journal and conference. His research interests include wireless sensor network and Internet of Things.

Tasher Ali SHEIKH received B.Tech degree in Electronics and Communication Engineering from Central Institute of Technology, Kokrajhar Assam, India, in 2013, M.Tech in Mobile Communication and Computing from National Institute of Technology, Arunachal Pradesh, India, in 2015, and Ph.D. from North Eastern Regional Institute of Science and Technology, Arunachal Pradesh, India in January, 2021, under the Visvesvaraya Ph.D. scheme under Ministry of electronics and Information Technology (Meity), Govt. of India. He is currently working as Project Coordinator at the National Institute of Electronics and Information Technology (NIELIT), Bhubaneswar, Odisha 751013, India. He has more than 4+ years of experience in research and teaching field. He has published 46 research articles in international referred journals and conferences. His field of interest includes massive MIMO, cell free massive MIMO, IoT and cell free IoT.

Nabajyoti MAZUMDAR received B.E degree in Information Technology from the University of Burdwan, India. He did his M.Tech and Ph.D. in Computer Science and Engineering from Indian Institute of Technology (ISM) Dhanbad, India, in 2014 and 2018 respectively. Currently, he is an Assistant Professor in the Department of Information Technology, Indian Institute of Information Technology Allahabad, India. He has acted as referees in many reputed journals including IEEE Internet of Things, IEEE Transactions on Industrial Informatics, IEEE System Journal, Simulation Modelling Practice and Theory, International Journal of Communication Systems etc. His current research interest includes wireless sensor network, IoT, and soft computing.

Amitava NAG (corresponding author) is currently a Professor of Computer Science and Engineering at the Central Institute of Technology Kokrajhar, Assam, India. He has more than 70 research publications in various international journals and conference proceedings. His research interests include IoT, information security and machine learning. He is a senior member of IEEE and a fellow of IEI.