Trajectory Design of UAVs-Assisted Edge Computing Systems for Efficient Data Collection from Animal Herds

Nao Maeda*, Tomotaka Kimura†, Kouji Hirata‡
* Graduate School of Science and Engineering, Kansai University, Japan
† Faculty of Science and Engineering, Doshisha University, Japan
‡ Faculty of Engineering Science, Kansai University, Japan
E-mail: {k284448, hirata}@kansai-u.ac.jp, tomkimur@mail.doshisha.ac.jp

Abstract—**In this paper, we propose a trajectory optimization design method for an Unmanned Aerial Vehicle (UAV)-assisted edge computing system for mobile IoT clients attached to animals. In this system, the UAV approaches the client node to perform data collection and processing. However, due to the clients' mobility, it is challenging for the UAV to maintain accurate knowledge of their location. To address this, the proposed method aims to maximize communication opportunities while minimizing UAV movement under limited observability. In the proposed method, hierarchical clustering based on the spatial proximity of client nodes is applied to form clusters where some client nodes are selected as representatives. The UAV dynamically plans its trajectory using only the positions of these representatives at each time step. Through numerical experiments, we demonstrate that the proposed method enables efficient data collection in dynamic environments where animals move according to the Boids model, even in the absence of complete location information for all client nodes.**

I. Introduction

In recent years, advancements of Internet of Things (IoT) technologies have enabled a wide range of practical services, where collected data are typically processed by cloud data centers [1]. However, as the volume of data generated by IoT applications continues to grow rapidly, cloud-based systems have faced challenges, including communication congestion and latency, which degrade real-time performance.

To address this issue, edge computing has attracted considerable attention as a complementary approach to cloud computing [2]. In particular, UAV-assisted edge computing systems have been proposed as an effective solution for environments with limited communication infrastructure [3]–[5]. In such systems, UAVs equipped with onboard processors collect and process data from clients such as IoT devices that are located within their communication range. In the UAVs-assisted edge computing systems, UAVs are capable of freely flying within the designated area to perform data collection. However, due to their limited communication range, the UAVs need to approach client nodes closely enough to ensure reliable communication. Therefore, trajectory optimization of UAVs is essential for efficient data collection.

Additionally, UAV-assisted systems have recently been considered for applications such as livestock monitoring in large natural environments, including pastures and grazing lands [6]. In these scenarios, UAVs collect data from animals equipped with IoT devices to support health monitoring and to detect incidents such as accidents and straying. However, due to the wide-ranging and dynamic movement of animals [7], static trajectory planning is insufficient. To ensure reliable communications, a trajectory design method that can adapt to dynamic changes over time is required.

In this paper, we propose a UAV trajectory optimization method that enables efficient communication even under the constrained condition where the UAV does not have accurate knowledge of all client locations but knows the location of some representative clients. Introducing such constraints is expected to prolong the lifetime of battery-powered IoT devices through power-saving operation. The proposed method aims to maximize communication opportunities while minimizing UAV movement under limited observability. To do so, in the proposed method, hierarchical clustering based on the spatial proximity of client nodes is applied to form clusters where some client nodes are selected as representatives. We consider a time step to represent the movement of animal herds. At each time step, only the updated positions of the representative clients are obtained, and the UAV dynamically optimizes its visiting order of the clients accordingly. At each visit, the UAV communicates not only with representative clients but also with other nearby client nodes within its communication range. This approach enables flexible and efficient data collection in scenarios where the positions of all client nodes cannot be fully observed.

II. System model

Fig. 1 illustrates the system model assumed in this paper. A set $\mathcal{N}$ of client nodes (i.e., animals equipped with IoT devices) that moves within a closed two-dimensional area. The positions of clients are updated periodically, and the update interval is defined as a time step Δt. The movement of animals is modeled using the Boids model [8], in which the animals form some herds that are dispersed geographically. In each animal herd, n_r client nodes are randomly selected as representative clients.

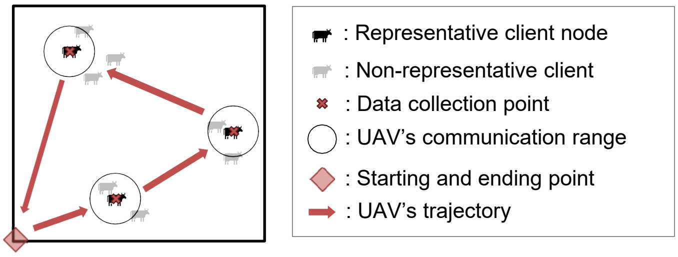

Fig. 1: System model.

A UAV flies within the area, with its communication range defined by a radius R. The UAV travels at a constant speed v to visit client nodes and completes its mission by returning to its initial departure point. The UAV determines its trajectory based on the latest positions of the representative client nodes and visits them sequentially to carry out data collection and processing. When the UAV reaches the position of a representative client node, it hovers for a fixed duration t_c to perform data collection and processing. At the same time, data processing is performed not only for the representative client node but also for all client nodes within the communication range.

The objective of this paper is to dynamically determine the UAV's visiting order and trajectory based on limited information from representative client nodes, in order to minimize travel distance while maximizing data processing opportunities.

III. Proposed Method

A. Overview

The proposed method first performs hierarchical clustering based on the positions of the client nodes in order to identify animal herds. Each animal herd is treated as a cluster. In each cluster, n_r representative client nodes are selected to effectively determine the communication targets. Subsequently, at each time step, the UAV dynamically determines the visiting order of representative client nodes by solving the Traveling Salesman Problem (TSP) based on their latest positions. The UAV then visits the representative client nodes sequentially and performs data processing. This series of processes is repeated until the UAV completes the data processing of all representative client nodes, enabling flexible and efficient trajectory planning even under limited positional information.

B. Hierarchical clustering

The proposed method sequentially designs the UAV's trajectory to efficiently collect data from mobile client nodes, under the constraint that the position of only representative client nodes in each animal herd can be obtained. To do so, the proposed method first performs hierarchical clustering based on the positions of the client nodes in order to identify animal herds that are dispersed geographically. In this paper, we use the Complete-Linkage algorithm for hierarchical clustering as follows.

1) Each client node is initialized as an individual cluster.

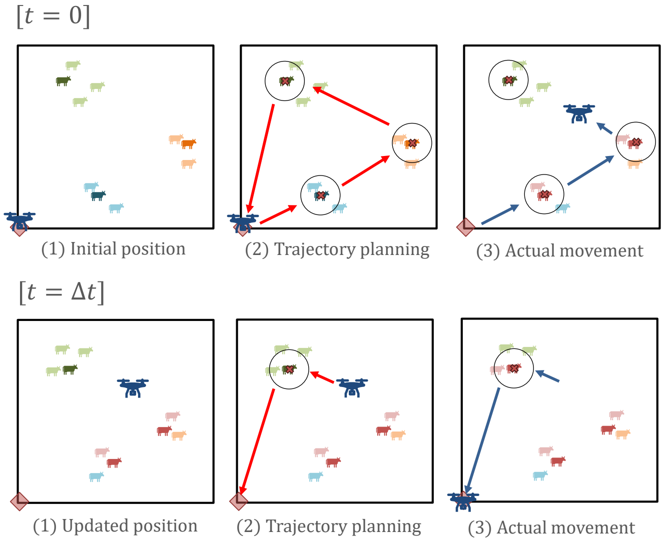

Fig. 2: Procedure of the proposed method.

2) For each pair of clusters (A, B), it computes the distance:

$$D(A,B) = \max_{a \in A, b \in B} d(a,b), \tag{1}$$

where $d(a,b)$ is the distance between clients a and b.

3) It finds the pair of clusters with the smallest value of $D(A,B)$.

4) If $D(A,B) > \tau$, the algorithm terminates; otherwise, it merges them into a single new cluster and then goes back to 2), where τ is a parameter that indicates the maximum allowable size of clusters.

By using this algorithm, clusters are formed so that the maximum distance between any pair of nodes within a cluster does not exceed τ. Then, n_r client nodes are randomly selected in each cluster to construct the representative client node set. In the subsequent time steps, the positions of only these representative client nodes are used for trajectory planning.

C. Trajectory planning

Fig. 2 illustrates an example of the trajectory planning in the proposed method. Assuming that client node positions are updated every time step Δt, the UAV acquires the latest locations of representative client nodes at each period and updates its trajectory accordingly. In each period, the UAV determines the optimal visiting order by solving the TSP, and visits them sequentially within the time available for that period. By recalculating the UAV's trajectory based on the latest available information and solving the TSP at every time step, the UAV can adaptively follow the moving targets while enhancing data processing efficiency and minimizing travel distance. The detailed procedure of the trajectory planning in the proposed method is as follows.

1) The current positions of unvisited representative client nodes are obtained.

2) The TSP using these positions is solved to determine the optimal visiting order.

TABLE I: System parameters for the Boids model.

Parameter	Symbol	Value
Simulation area	–	2000×2000 [m]
Number of client nodes	$\lvert\mathcal{N}\rvert$	100
Speed of client nodes	v_i	$25 \times (0.8$ to $1.2)$ [m/s]
Number of simulation steps	S	180
Time per step	Δt	30 [s]
Neighborhood radius	r_n	200 [m]
Alignment weight	w_a	6.0
Cohesion weight	w_c	1.5
Separation weight	w_s	1.0
Global randomness factor	η	0.8
Individual randomness factor	η_i	0.5 to 1.0

3) As long as the total travel and processing time remains within the time step Δt, the following steps a)-c) are performed.
   a) The UAV visits the representative client node according to the visiting order.
   b) The UAV hovers for a fixed duration t_c to perform data processing for all clients within its communication range R.
   c) If the visit to all representative client nodes is completed, the UAV returns to its origin, and the procedure terminates; otherwise, it goes back to a).
4) If the remaining time is insufficient to visit the next client node, the UAV suspends movement. Then, it proceeds to the next time step and goes back to 1).

IV. Performance Evaluation

A. Client node mobility model

To evaluate the proposed method, we simulate the movement of client nodes using the Boids model [8], which reproduces natural flocking behavior through simple interaction rules among agents. In the simulation experiments, $|\mathcal{N}| = 100$ client nodes move within a 2000×2000 [m] area for $S = 180$ time steps, with each step lasting $\Delta t = 30$ [s]. Client nodes are initially placed randomly and exhibit flocking behavior such as alignment, cohesion, and separation. Only data from time steps $S = 60$ to 180 are used for evaluation, during which meaningful group dynamics are established. Specifically, data from early time steps (i.e., $S < 60$), during which the animals have not yet formed herds due to the small number of steps, are excluded from the evaluation. The system parameters for the Boids model used in the simulation experiments are summarized in Table I.

The communication range R of the UAV is set to 100 [m]. The flight speed v of the UAV is set to 10 [m/s]. The hovering duration t_c of the UAV at each representative client node is set to 3 [s]. The starting and ending point of the UAV is set to $(0, 0)$. The system parameters for the UAV are summarized in Table II.

B. Results

Fig. 3 shows an example of client node distributions at $S = 60$ before UAV starts data processing, where the number n_r of representative client nodes in each cluster is set to 1. As we can see from this figure, animals form herds, and the proposed method constructs clusters accordingly. Fig. 4 illustrates the UAV trajectory after the UAV completes data processing for all representative nodes. These figures illustrate how the UAV trajectory and client node distribution change before and after data processing. The UAV sequentially visited representative client nodes and successfully communicated with multiple client nodes located within its communication range at each hovering point, demonstrating efficient data processing behavior.

TABLE II: System parameters for the UAV.

Parameter	Symbol	Value
UAV communication range	R	100 [m]
UAV flight speed	v	10 [m/s]
Hovering duration	t_c	3 [s]
UAV start/end location	-	(0, 0)
# of representative client nodes per cluster	n_r	1 to 9, or All
Clustering distance threshold	τ	300 [m]

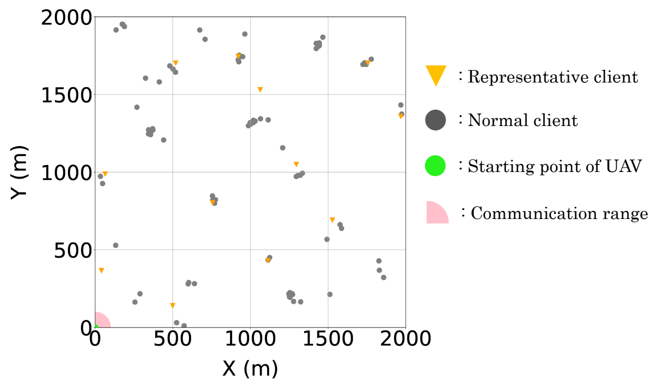

Fig. 3: Client node distribution.

Subsequently, we quantitatively evaluate the performance of the proposed method as the number of representative client nodes in each animal herd is varied. We conduct 10 independent trials to show the average performance.

Fig. 5 shows the coverage rate of client nodes as a function of the number n_r of representative client nodes in each animal herd. The coverage rate of client nodes is defined as the proportion of client nodes that were processed within the UAV's communication range concurrently with the representative client nodes. We also plot the result (labeled with All) in the case where all client nodes are selected as representative client nodes. From this figure, we observe that increasing the number n_r of representative client nodes enhances the coverage rate. We also observe that more than 90% of the client nodes are covered when $n_r \geq 4$.

Fig. 6 shows the total flight time as a function of the number n_r of representative client nodes in each animal herd. As shown in this figure, the total flight time increases with the number n_r of representative client nodes. This is because an increase in n_r requires the UAV to visit more nodes.

These findings demonstrate that the number of representative

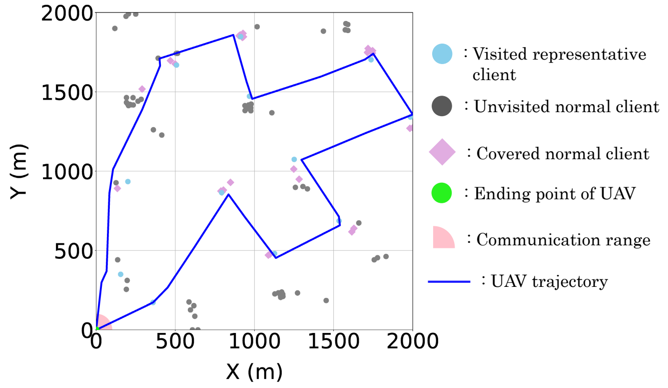

Fig. 4: UAV trajectory example.

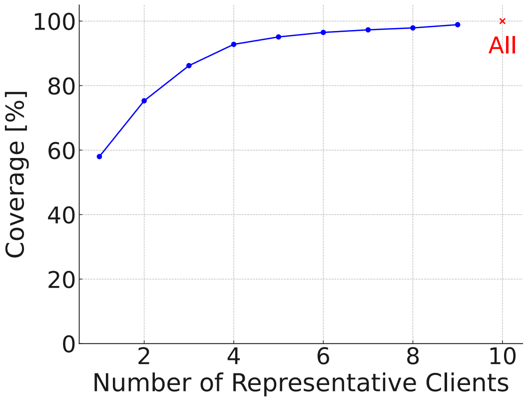

Fig. 5: Coverage rate.

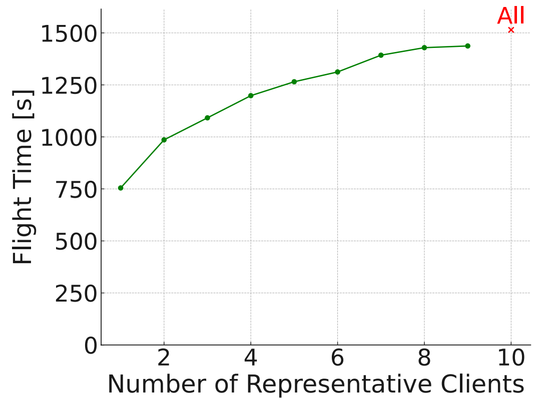

Fig. 6: Total flight time.

client nodes is a key parameter that controls the trade-off between the coverage rate and the UAV's movement cost. Therefore, selecting an appropriate number of representative client nodes is essential for system designers to balance data processing efficiency and UAV resource constraints under varying environmental conditions.

V. Conclusion

In this paper, we proposed a trajectory optimization design method for UAV-assisted edge computing systems, in which representative client nodes are selected and their updated positions are used for dynamic UAV trajectory planning. Simulation results based on the Boids model demonstrated that while increasing the number of representative client nodes enhances the coverage rate, it also increases the UAV's flight time, revealing a trade-off between data processing efficiency and UAV movement cost. The proposed method effectively balances this trade-off by selecting an appropriate number of representative client nodes in dynamic environments. Future work includes extending the framework to multiple UAVs and incorporating additional system constraints.

Acknowledgment

This research was partially supported by JSPS KAKENHI Grant No. 23K11077 and No. 25K15103.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] X. Sun and N. Ansari, "Edgeiot: Mobile edge computing for the internet of things," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 22–29, 2016.

[3] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[4] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[5] P. McEnroe, S. Wang, and M. Liyanage, "A survey on the convergence of edge computing and ai for uavs: Opportunities and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 15 435–15 459, 2022.

[6] P. Chamoso, W. Raveane, V. Parra, and A. González, "UAVs applied to the counting and monitoring of animals," in *Ambient Intelligence-Software and Applications: 5th International Symposium on Ambient Intelligence*, Springer, 2014, pp. 71–80.

[7] D. W. Bailey, "Grazing and animal distribution," in *Animal Welfare in Extensive Systems*, Sheffield: 5M Publishing, 2016, pp. 53–77.

[8] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, New York, NY, USA, 1987, pp. 25–34.