Abstract—Due to the sustainable power supply and environment-friendly features, self-powered IoT devices have been increasingly employed in various fields such as providing observation data in remote areas, especially in rural areas or post-disaster scenarios. Generally, through multi-hop relay, the sensed data of those self-powered IoT devices are collected by the sink node which connects to the IoT backbones. However, due to the remoteness, the sink needs to be located at the border of the monitoring area where both the backbone of IoT and electrical infrastructures are accessible. Under such deployment, significant data flow and relay overhead will incur considering the large scale of the monitoring area. Motivated by this issue, this paper aims to design a UAV-assisted self-powered heterogeneous system to provide comprehensive monitoring data. In this system, because of the superiority of the unmanned aerial vehicle (UAV) in the easy deployment, We dispatch UAV collects data from self-powered IoT devices, periodically so as to alleviate the data overflow. Moreover, based on that the self-powered IoT devices are expected to have a more considerable capability in the heavy data flow area, we also developed a placement upgrade strategy to upgrade the general homogeneous self-powered IoT system to the heterogeneous self-powered IoT system. Simulation results indicated the developed UAV-assisted self-powered heterogeneous system can achieve around \(1.28\times\) the amount of data delivery to sink compared with the homogeneous self-powered IoT system.

I. INTRODUCTION

In the past twenty years, IoT devices are increasing at an incredible rate. IoT devices are expanding to every corner of our world [1, 2]. Those sensors observe our earth and then transmit their full-dimensional and comprehensive observation to the data center through multi-hop routing assisting various decision making, which benefits our daily life. For instance, [3–5] NASA launched “Sensor Web” project to utilize sensors to create an electronic skin of our earth, where enormous sensors 24/7 monitor environment such as flooding, volcano event, and soil humidity. However, due to the battery limited lifetime, IoT devices work with a limited lifespan. The replacement either on such billions of IoT devices or their batteries results in huge costs. To extend the IoT device’s lifetime, a self-powered IoT device is proposed that harvests energy from natural sources to sustainably power the various onboard operations of IoT devices such as sensing and communication.

Nevertheless, due to the weak and unstable natural power supply of self-powered IoT devices, the self-powered IoT devices might be frequently interrupted during routing data packets to other devices, which results in reconnection thus dissipating energy. Therefore, the self-powered IoT devices have to consider the onboard energy status of the transmitter and receiver while routing data. Unlike the self-powered IoT device, because the general IoT devices have a stable power supply so that they can work continuously and keep the connection between transmitter and receiver, the conventional IoT communication developed for the general IoT devices does not consider such intermittent work pattern features. Thus, conventional communication is not appropriate for self-powered IoT devices. An optimal routing policy is on-demand for self-powered IoT devices. Because limited onboard energy is allocated not only to sense but also to transmission and receiving, thus, to obtain the optimal routing policy the energy allocation should be optimized with routing policy together.

In this background, we have investigated the multi-hop routing and energy allocation for self-powered IoT systems in our previous study [6], [7], where a set of self-powered IoT devices are deployed to monitor the environment and transmit the collected data to the data center. We maximized the amount of data delivery to the data center through jointly optimizing the multi-hop routing and energy allocation with the multi-agent deep reinforcement learning, named GAP. In this paper, the data center is located at the center of the deployment area. This paper extends it to a more specific scenario, where the data center is located at the edge of the deployment area and only the self-powered IoT devices that are close to the sink can directly reach out to the data center. Other devices only can transmit data packets through multi-hop routing, as shown in Fig. 2(a). Such a system is widely applied in many scenarios. For instance, a set of IoT devices is deployed to monitor environmental parameters after a disaster such as an earthquake or provide environment observation in a rural area or military area, where generally the backbone of IoT cannot be reachable for those far away IoT devices.

Although such a scenario is widely encountered in real life, routing all needed data to the edge data center is extremely challenging. As Fig. 2(a) indicated all sensed data of the whole self-powered IoT system are routed via the IoT devices that are directly linked with the sink (one-hop neighbors of the sink). However, those self-powered IoT devices also have limited energy, which caused some sensed data can not be routed resulting in data overflow. The detailed explanation and mathematical analysis of data overflow is given in Section III. To alleviate data overflow, we leverage the UAV’s mobility,
flexibility, and easy-deployment merits dispatching a UAV to collect sensed data from a portion of IoT devices. Even if the UAV is dispatched to collect data, the data collection efficiency is degraded. This is because, while UAVs collect data from a device, this device is regarded as a data aggregation station to route the data packets from its around devices to UAV. Similar to the devices directly linked with the sink, because of the weak power of this “data aggregation station”, the self-powered IoT devices might not have sufficient energy to transmit all the data packets to UAV while UAV coming for collection.

Therefore, instead of building up a homogeneous self-powered IoT system deploying all IoT devices with the same capability, in this paper, we design a heterogeneous self-powered IoT system as Fig. 2(b) where a portion of self-powered IoT devices are upgraded to master IoT devices that has a larger energy harvester and storage. UAV will be dispatched to collect data from those master IoT devices. In this paper, we first inherit a multi-hop routing policy from the homogeneous self-powered IoT environment monitoring system from [7]. To build up a UAV-assisted heterogeneous self-powered IoT system, we developed the Max-Cover algorithm to decide the master IoT device selection from the homogeneous self-powered system, where which ones the regular IoT devices should be upgraded to the master IoT devices are selected. After that, UAV is dispatched to collect data from a portion of master IoT devices with energy constraints. Through optimizing the master IoT devices selection and the UAV trajectory planning, our goal is to maximize the amount of data delivery to the sink.

The rest of this paper is organized as follows. We first briefly review the related work in Section II. Then we show a motivation example to illustrate the overflow data case in Section III. After that, we build up our UAV-assisted hybrid IoT system Section IV. We illustrate the Radius-Wise UAV trajectory planning in Section V whose performance is evaluated through the simulation experiments given in Section VI. Finally, we conclude this paper in Section VII.

II. RELATED WORK

In very recent years, UAV-assisted data collection has attracted interest both from academics and industry. Motivated by the flexibility, mobility, and easy-deployment merits, UAV-assisted data collection is widely applied in various fields such as soil humidity detection and volcano monitoring. In [8], IoT devices are distributed aside from highway collecting data. UAV is dispatched to each IoT device and collects data from those devices while hovering at each device. This paper mainly targets highway scenarios. While the IoT devices are not distributed on a line, the explored method is not appropriate. [9]. [10] researched UAV flies to each ground sensor node then hovers to collect data, where the age of information collected is maximized through optimizing the UAV path with dynamic programming, genetic algorithm, and ant colony heuristic algorithm. However, the UAV constraint did not take into account. Thus, it is more suitable for small-scale scenarios. In [11], all devices are clustered based on the locations of IoT devices, and then UAV flies to each cluster head to collect data. This scenario did not consider a multi-hop routing for the ground IoT devices. Moreover, all UAV-assisted data collection system is constructed by the general IoT devices but there is no study target on UAV to assist self-powered IoT devices. Therefore, this paper aims to construct a UAV-assisted self-powered IoT environment monitoring system.

III. MOTIVATION ANALYSIS

Fig. 1 indicates a small example of the homogeneous self-powered system. Each device senses data with $v_s$ bit/s sensing rate. All IoT devices transmit the sensed data to Sink through multi-hop routing. Define the power of sensing, transmitting, and receiving for each node are $p_s$, $p_t$, and $p_r$ (J/bit), respectively. Given the operation time of the self-powered IoT system as $T$, we can calculate the energy consumption as follows.

$$E_{1, \text{cost}} + E_{2, \text{cost}} = 2v_sT_p + 6v_sT_p + 4v_sT_p$$  \hspace{1cm} (1)
$$E_{3, \text{cost}} + E_{4, \text{cost}} = 2v_sT_p + 4v_sT_p + 2v_sT_p$$  \hspace{1cm} (2)
$$E_{5, \text{cost}} + E_{6, \text{cost}} = 2v_sT_p + 2v_sT_p$$  \hspace{1cm} (3)

where (1) represents the energy costs of device1 and device2 (devices linked to sink with one-hop). That energy cost includes three components, sensing energy costs of device1 and device2, transmission energy costs that device1 and device2 transmit all data packets sensed by 6 devices to Sink, and receiving energy costs that device1 and device2 receive the data packets from device3 and device4 that is sensed by device3 to device6 4 devices. Similarly, we have (2) and (3), which describes energy costs of devices linked to sink with two-hops and three-hops, respectively.

Given the onboard harvester parameters and the power intensity, we define the maximum energy that can be utilized for period operation time $T$ as $E_{T,U}$. To guarantee that the self-powered system run with sufficient energy, $2E_{T,U} \geq E_{1, \text{cost}} + E_{2, \text{cost}}$, $2E_{T,U} \geq E_{3, \text{cost}} + E_{4, \text{cost}}$, and $2E_{T,U} \geq E_{5, \text{cost}} + E_{6, \text{cost}}$ to have to be satisfied. However, as (1) to (3) indicated, there is a difference between $E_{1, \text{cost}} + E_{2, \text{cost}}$ and $E_{5, \text{cost}} + E_{6, \text{cost}}$. If we select harvester based on the value of $E_{1, \text{cost}} + E_{2, \text{cost}}$, the harvesters resource of device5 and device6 are dissipated. If we select harvester based on the value of $E_{5, \text{cost}} + E_{6, \text{cost}}$, the device1 and 2 will not have sufficient energy to complete the data routing task, which causes that the sensed data get stuck and can not be delivered to Sink. We call this situation as data overflow.

Therefore, instead of deploying devices homogeneously in the whole deployment area, to realize cost-effective deployment, we propose to deploy self-powered devices heterogeneously, where the devices have a more considerable capability on energy harvesting and storage in some specific locations. To alleviate data overflow, UAV is dispatched to collect data. The details of the system model and algorithm design are described in Section IV and Section V.
IV. SYSTEM MODEL & PROBLEM FORMULATION

A. System Model

In this paper, we consider the typical scenario where a set of self-powered IoT devices are deployed to sense data and then transmit the sensed data to Sink. With regarding the long distance among devices and Sink, the backbone of IoT devices only can be accessible for IoT devices that are close to Sink, as Fig. 2(a) shown. Thus, all sensed data packets are transmitted to Sink via multi-hop routing. Such scenario is frequently encountered in real life such that the devices detect parameters in vast forests, collect data in rural areas, or works in military areas that generally are not connected to network service.

![Diagram](image.png)

Fig. 2: Configuration of (a) the homogeneous self-powered system and (b) the UAV-assisted heterogeneous self-powered system.

We define the set of self-powered IoT devices as \( \mathcal{I} = \{1, \ldots, i, \ldots, I\} \). The location of device \( i \) is denoted as \((x_i, y_i)\). Because of the problems analyzed in our motivation example, we will increase a portion of self-powered IoT devices’ energy harvesting and storage capability, called those devices as master IoT devices, denoted as \( \mathcal{M} = \{1, \ldots, m, \ldots, M\}\). The configuration of heterogeneous self-powered system is as Fig. 2(b) shown. Given the transmission range as \( \xi \), if the distance between two devices is within the transmission range, two devices are neighbors for each other and can route the data packet with one-hop. The neighbors set of \( i \) is \( N_i \). The multi-hop routing policy and energy allocation policy of the heterogeneous self-powered system inherits from paper [7]. The sensing, transmitting, and receiving power of each device are \( p_s, p_t, \) and \( p_r \), respectively. These three operations are powered by the harvested energy, labeled as \( E_{i, \text{havst}}(E_{m, \text{havst}}) \). Given the power trace as \( P_T = \{p_1, \ldots, p_T\} \) where \( p_i \) is the power intensity at time \( t \), the harvested energy from \( t_1 \) to \( t_2 \) is calculated by \( E_{i, \text{havst}} = \int_{t_1}^{t_2} p_i dt \). Define the maximum onboard energy of \( i \) and \( m \) as \( E_{i, \text{max}} \) and \( E_{m, \text{max}} \), respectively. The real time energy of \( i \) and \( m \) at time \( t \) are labeled as \( E_{i, t} \) and \( E_{m, t} \). \( E_{i, t} \leq E_{i, \text{max}} \) and \( E_{m, t} \leq E_{m, \text{max}} \) has to be satisfied.

To collect the data effectively and alleviate the data overflow, the UAV \((d)\) mounted with a single antenna is dispatched to collect data from master IoT devices. The UAV departs from the docking station located at \( D \) and then flies to the selected master IoT devices and hovers at each master IoT device for data collection. Finally, UAV returns back to the same docking station after the mission is completed. For simplicity, we assume UAV flies at altitude \( H \) with a constant speed \( v_u \). We denote the maximum onboard energy used for propulsion is \( E_{u, \text{max}} \). Due to the limited onboard energy of the UAV, UAV might return to the docking station for recharging or change its battery for the next flight. Moreover, the self-powered IoT devices need duration to sense data and aggregate the sensed data to the master IoT devices. Therefore, the UAV is dispatched to perform a collection mission per \( t_f \) time. Each time the maximum energy used for propulsion is \( E_{u, \text{max}} \). To mathematically formulate our problem, we label the feasible UAV path as \( U = \{u_1, \ldots, u_k, \ldots, u_K\} \), where \( u_k \) is the location of \( k \)-th visited master IoT device. At each visited master IoT device, the UAV hovers for \( t_h \) time. This paper dispatches a rotary-wing UAV, whose propulsion power is determined by the flight velocity. Define the propulsion power is \( p_u(v_u) \). Thus, the propulsion energy cost of UAV \( E_u \) is given by \( (4) \), based on [12].

\[
E_u = p_u(v_u) \frac{d_{a,b}}{v_u} + p_u(v_u) \sum_{k=1}^{K-1} \frac{d_{u_k,u_{k+1}}}{v_u} + p_u(v_u) \frac{d_{a,b}}{v_u} + p_u(0) t_h K
\]

where the \( d_{a,b} \) represents the Euclidean distance between the waypoints \( a \) and \( b \), \( p_u(0) t_h K \) is the energy cost on the hovering \((v_u = 0)\). Because the energy consumption on data receiving is small compared with the propulsion energy consumption, the receiving energy consumption is ignored. Thus, \( E_u \leq E_{u, \text{max}} \) has to be satisfied.

B. Problem Formulation

Our goal is to maximize the amount of data delivery to Sink and UAV through optimizing the master IoT device selection and UAV trajectory optimization. Denote \( A_{a,b,t} \) as the size of data packet that \( b \) received from \( a \) at time \( t \). The mathematical formulation is as follows:

\[
\begin{align*}
\text{maximize} & \sum_{i=t} \sum_{t=1}^T A_{i,t,Sink,t} + \sum_{i=t}^T A_{i,t,\text{max}} \\
\text{subject to} & 0 \leq E_{i,t} \leq E_{i,\text{max}}, \quad \forall i \in \mathcal{I}, t \leq T \\
& E_u \leq E_{u,\text{max}} \\
& M \in \mathcal{I} \quad \& \quad \forall u_k \in \mathcal{M}
\end{align*}
\]

V. THE MAX-COVER ALGORITHM

The Max-Cover algorithm designs a heterogeneous self-powered IoT system through three phases: 1) Select the master IoT devices \( \mathcal{M} \) from \( \mathcal{T} \); 2) Select the master IoT device that UAV will visit in \( \mathcal{M} \); 3) Figure out the visited order of those selected master IoT devices so that work out the UAV trajectory planning. To complete these three phases, our key idea is first to prune IoT devices that are unlikely to be a master IoT device from the set of all IoT devices \( \mathcal{I} \), such as the devices located at the marginal deployment area. This can narrow the master IoT device search space and obtain the candidates of master IoT devices. After that, given the maximum number of master IoT devices \( \mathcal{M} \), we select the master IoT devices from the candidates of master IoT devices by maximizing the number of one-hop neighbors of all master IoT devices. Finally, we
Fig. 3: Visualization of (a) the Pruning edge device, (b) the Pruning last layer device, and (c) the definition of one-hop coverage.

decide the visited order of master IoT devices by the iterative genetic algorithm.

A. Pruning on the candidates of Master IoT devices

Assume that all IoT devices in $\mathcal{I}$ are the candidates of master IoT devices at first, which means we initialize $\mathcal{M} = \mathcal{I}$. To eliminate the devices that are unlikely to be a master IoT device from the candidate set $\mathcal{M}$, we prune two types of IoT devices from $\mathcal{M}$. Firstly, we prune the IoT devices located at the marginal deployment area. Based on the experiment analysis in [7], the IoT devices located at the marginal deployment area mainly function as “sensing” nodes without routing data packets for others because of their positions. However, master IoT devices as a data aggregation station aim to route data packets to the UAV. Thus, the IoT devices located at the marginal deployment area are not appropriate to be master IoT devices. We define those devices as the marginal IoT devices denoted as set $\Delta$, $\Delta \in \mathcal{I}$, which is found as Fig. 3(a) indicated. Given the positions of devices, each device regards as one point. $\Delta$ is the convex full vertex of the set $\mathcal{M}$. We adopts the Graham algorithm [13] to determine $\Delta$. The algorithm is described as Algorithm 1.

Secondly, based on \[1\] to \[3\], the devices that are farther away from the sink, the workload is smaller. Therefore, those devices are not expected to be master IoT devices. But the devices that are near to the sink have a heavy workload. Thus, those devices could be the master IoT devices candidates. In this situation, we first separate the deployment area into multiple layers as Fig. 3(b). The center of the layer boundary circle is located at the sink. The width of each layer is the transmission range of self-powered IoT devices. Defined the devices located at the final layer as $\mathcal{F}$ and the devices located at the first layer as $\mathcal{Z}$. We discard those marginal devices in $\Delta$ and $\mathcal{F}$ but keep the devices in $\mathcal{Z}$. Thus, $\mathcal{M} = \mathcal{I} - \Delta - \mathcal{F} + \mathcal{Z}$.

Define $|\mathcal{M}|$ as the cardinality of set $\mathcal{M}$. In the pruning step, we narrowed down the set of $\mathcal{M}$. Given the maximum number of master devices as $|\mathcal{M}|$, after pruning, if $|\mathcal{M}| \geq |\mathcal{M}|$, we let the final set of master IoT devices be $\mathcal{M}$. While $|\mathcal{M}| < |\mathcal{M}|$ after pruning, it means we should further select the master IoT device from current $\mathcal{M}$. At this time, we maximize the number of one-hop neighbors that master IoT devices could have in $\mathcal{M}$. Recall that $N_i$ is the one-hop neighbor of $i$ device. We will select $\mathcal{M}$ devices from $\mathcal{M}$ that lead to the maximum $|N_1 \cup N_2 \cup \cdots \cup N_{|\mathcal{M}|}|$. It implies that the one-hop coverage range of master IoT devices is maximized, as Fig. 3(c) shown. Specifically, we maximize $|N_1 \cup N_2 \cup \cdots \cup N_{|\mathcal{M}|}|$ which avoids counting the number of devices located in shadow area of Fig. 3(c) repeatedly, which means those devices are the one-hop neighbor of several devices simultaneously.

B. UAV Trajectory Planning

Through pruning and maximizing the one-hop neighbor number of all master IoT devices, we selected the master IoT device $\mathcal{M}$ from $\mathcal{I}$. However, because of two factors, not each master IoT device can be visited. The first reason is that the UAV has limited onboard energy leading to the limited flight distance and duration. The second is that the master devices at the first layer in Fig. 3(b) are linked to the sink directly. Those devices can transmit their data packets to sink with one-
hopping, instead of utilizing the UAV collection. Therefore, while planning UAV trajectory, UAV will not visit the devices in the first layer area, which means the elements of \( U \) are in the set \( M - Z \).

To decide the UAV trajectory, we adopt the genetic algorithm (GA) as the underlying algorithm \([4]\). We have two tasks: 1) Select which devices can be visited in \( M - Z \); 2) Decide the visited order of the selected devices in 1). In our algorithm, we define each individual consists of several genes, where each gene represents a master device. The order of gene represents the visited order. The fitness of each individual is evaluated through the weighted sum of travel distance score and distribution score. The travel distance score is defined as 

\[
\sum_{k=1}^{K} d_{u_k, u_{k+1}} + d_{u_K, D},
\]

where \( D \) is equal to \( D = d_{D, u_1} + \sum_{k=1}^{K-1} d_{u_k, u_{k+1}} + d_{u_K, D} \) that is the total distance UAV travelled. The distribution score is 

\[
D = \sum_{k=1}^{K} (d_{u_k, u^*}),
\]

where \( u^* \) is the centroid \( U \). Specifically, when the value of \( H \) is smaller, the visited master IoT devices are concentrated, which is unfriendly for routing. Therefore, mathematically the fitness of each individual is \( \alpha D + \beta H \), where \( \alpha \) and \( \beta \) are the weight factors on the distance and distribution score, respectively. The genetic algorithm maximizes the value of \( \alpha D + \beta H \) through searching different trajectories. In particular, to satisfy the energy-constrained condition, once the individual is generated, if the trajectory energy cost exceeds the energy-constrained \( E_{i, max} \), the individual will be discarded. If the algorithm cannot find out the feasible trajectory, we will decrease one gene number to perform the algorithm again each time until finding the feasible UAV trajectory with the maximum fitness. Algorithm \([5]\) indicates the implementation of UAV trajectory. \( U^* \) labels the optimal UAV trajectory.

VI. SIMULATION RESULTS

We developed a self-powered IoT network simulator to evaluate the designed UAV-assisted heterogeneous self-powered environment monitoring system in this section. All IoT devices harvest energy from solar power which is downloaded from \([15]\).

A. Experiment Settings

Environment monitoring system. We simulate our environment monitoring system as Fig. 2. The system starts monitoring the environment from 8:00 to 18:00 on each training day. For fairness comparison and ensuring sufficient start-up energy, at the beginning of each training day, the initial energy of each device is equal to \( E_{i, max} \). DRL-based multi-hop routing policy is trained for 80 days, where each day represents one episode. The system parameters list in Table 1. The related parameters on DRL training can be retrieved from our paper \([7]\).

UAV Parameter. Given the duration of UAV as 15 mins, UAV flies to collect data per 30 mins, and at each visited master IoT device UAV hovers 60 seconds for collecting data from the corresponding master IoT device. UAV flies with 5m/s at 20-meter height. The trajectory optimization parameters are listed in Table 1.

Baselines. Because the problem this paper researched has not been studied in the existing works, there are no existing suitable methods that can be our baselines. Therefore, to evaluate the effectiveness of the proposed algorithm, we set our baselines as follows:

- **Baseline 1.** This baseline implements the homogeneous self-powered IoT system developed in \([7]\). The DRL-based multi-hop routing adopts DQN \([16]\) as underlying, instead of A3C. The homogeneous self-powered IoT system consists of one sink and 20 self-powered IoT devices whose topology is indicated in Fig. 4.
- **Baseline 2.** This baseline implements the heterogeneous self-powered IoT system, which is composed of one sink and 20 self-powered IoT devices including master self-powered IoT devices and general self-powered IoT devices, as Fig. 5. The exact number of master self-powered IoT devices and general self-powered IoT devices is discussed in the following parameter discussion section.

B. Parameter Discussion

To upgrade self-powered IoT devices, the parameter \( M \) needs to be configured. In order to figure out the impact of the number of master IoT devices, we vary \( M \) from 0 to 12 where the step length is 3. As the impacts of \( M \) reflects on

---

**Table I: Experiment Parameters**

| Notation & Definition | Value/Range |
|-----------------------|-------------|
| transmission range \( \langle \rangle \) | 20 |
| size of data packet | [3072, 7224] bits |
| transmission power \( P_{i,trans} \) | 0.1J |
| receiving power \( P_{i,recv} \) | 0.05J |
| sleeping power \( P_{i,slack} \) | 0.0005J |
| sensing power \( P_{i,sense} \) | 0.01J |
| energy capacity of \( i \) \( E_{i, max} \) | 1J |
| energy capacity of \( m \) \( E_{m, max} \) | 3J |
| The population of GA \( N_{pop} \) | 20 |
| The number of children in GA \( N_{child} \) | 12 |
| The weight of distance score \( \alpha \) | 0.4J |
| The weight of distribution score \( \beta \) | 0.6J |
the total number of data delivery to sink, in Fig. 6 we show the corresponding total amount of data delivery to sink node versus with the training day at different master nodes. Note that while the number of the master node is 0, the system is a homogeneous self-powered IoT system. While $M$ varies from 0 to 9, the amount of data delivery to sink increases with the number of master IoT devices increasing. However, while $M$ jumps from 9 to 12, the sink received data decreased. From a general perspective, the improvement of data delivery of the heterogeneous self-powered IoT system is very slight compared with the homogeneous system. Thus, even though we increase the capability of IoT devices, because of the limitation of topology, the performance of homogeneous self-powered IoT systems can not be improved, as analyzed in Section VII.

Thus, we also compare the total amount of data delivery to sink of UAV-assisted self-powered IoT devices and that of heterogeneous self-powered IoT devices, as Fig. 7 indicated. We can see with the UAV assistants, the sink can receive more data even while the number of master IoT devices in heterogeneous self-powered IoT systems is different. In Fig. 6, while the number of master IoT devices is equal to 6, the system achieves the best performance. When a UAV participates in a self-powered heterogeneous system, a 12-node system is the best choice. Specifically, the improvement through UAV is highest in 12-node master and 8-node general IoT systems.

C. Overall Performance

In this section, we measure the overall performance of baseline 1, baseline 2, and our method. Baseline 1 and baseline 2 complete 35.44Mb and 39.33Mb sink received data, respectively. The proposed UAV-assisted self-powered heterogeneous system completes 45.28Mb, which is 1.15× than the best performance of heterogeneous self-powered IoT system and 1.28× than that of homogeneous self-powered IoT system.

VII. CONCLUSION

This paper designs an UAV-assisted heterogeneous self-powered IoT system. We proposed the Max-Cover algorithm to generate the upgrade strategy for the homogeneous self-powered IoT system, where a partial of self-powered IoT devices are selected as master IoT devices whose capability are upgraded to alleviate the data overflow. After that, we dispatch UAV to collect data from those master IoT devices. The trajectory of UAV is optimized through genetic algorithm. Experiments indicate the improvement on data collection.

REFERENCES

[1] C. Pan, M. Xie, and J. Hu, “Enzyme: An energy-efficient transient computing paradigm for ultra-low self-powered iot edge devices,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 11, pp. 2440–2450, 2018.
[2] C. Pan, M. Xie, S. Han, Z.-H. Mao, and J. Hu, “Modeling and optimization for self-powered non-volatile iot edge devices with ultra-low harvesting power,” ACM Transactions on Cyber-Physical Systems, vol. 3, no. 3, pp. 1–26, 2019.
[3] S. Chien, B. Cichy, A. Davies, D. Tran, G. Rabideau, R. Castano, R. Sherwood, D. Mandl, S. Frye, S. Shulman et al., “An autonomous earth-observing sensorweb,” IEEE Intelligent Systems, vol. 20, no. 3, pp. 16–24, 2005.
[4] “Volcano sensorweb.” [Online]. Available: https://ai.jpl.nasa.gov/public/projects/sensorweb/.
[5] S. H. Liang, A. Croitoru, and C. V. Tao, “A distributed geospatial infrastructure for sensor web,” Computers & Geosciences, vol. 31, no. 2, pp. 221–231, 2005.
[6] W. Zhang, T. Liu, M. Xie, J. Zhang, and C. Pan, “Sac: A novel multi-hop routing policy in hybrid distributed iot system based on multi-agent reinforcement learning,” in 2021 22nd International Symposium on Quality Electronic Design (ISQED). IEEE, 2021, pp. 129–134.
[7] W. Zhang, T. Liu, M. Xie, L. Li, D. Kar, and C. Pan, “Energy harvesting aware multi-hop routing policy in distributed iot system based on multi-agent reinforcement learning,” in 2022 27th Asia and South Pacific Design Automation Conference (ASP-DAC). IEEE, 2022.
[8] G. J. C. T et al., “Flight time minimization of uav for data collection over wireless sensor networks,” IEEE Journal on Selected Areas in Communications, vol. 36, no. 9, pp. 1942–1954, 2018.
[9] L. J et al., “Age-optimal trajectory planning for uav-assisted data collection,” in IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2018, pp. 553–558.
[10] H. H et al., “Aoi-minimal trajectory planning and data collection in uav-assisted wireless powered iot networks,” IEEE Internet of Things Journal, vol. 8, no. 2, pp. 1211–1223, 2020.
[11] J.Liu, P.Tong et al., “Uav-aided data collection for information freshness in wsn,” IEEE Transactions on Wireless Communications, vol. 20, no. 4, pp. 2368–2382, 2020.
[12] Y. Zeng, J. Xu, and R. Zhang, “Energy minimization for wireless communication with rotary-wing uav,” IEEE Transactions on Wireless Communications, vol. 18, no. 4, pp. 2329–2345, 2019.
[13] R. L. Graham, “An efficient algorithm for determining the convex hull of a finite planar set,” Info. Pro. Lett., vol. 1, pp. 132–133, 1972.
[14] D. Whitley, “A genetic algorithm tutorial,” Statistics and computing, vol. 4, no. 2, pp. 65–85, 1994.
[15] “Measurement and instrumentation data center (midc).” [Online]. Available: https://midcdmz.nrel.gov/apps/sitehome.pl?site=ORNL.
[16] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double q-learning,” in Proceedings of the AAAI conference on artificial intelligence, vol. 30, no. 1, 2016.