applied sciences

MDPI

Article

Task Duration-Constrained Joint Resource Allocation and Trajectory Design for UAV-Assisted Backscatter Communication System

Wenxin Zhou and Long Suo *

College of Information Science and Technology & Artificial Intelligence, Nanjing Forestry University, Nanjing 210037, China; zwx20240507@163.com

* Correspondence: lsuo@njfu.edu.cn

Abstract

Backscatter communication (BackCom) has emerged as an energy-efficient and low-cost communication paradigm, in which wireless devices transmit information by reflecting incident signals rather than actively generating radio frequency signals. Owing to the extremely low power consumption and hardware cost, BackCom is particularly suitable for Internet of Things (IoT) devices with stringent low energy and cost constraints. However, due to the severe double channel attenuation inherent in backscatter links, conventional ground-based deployment of transmitters and receivers often suffers from poor communication quality and low energy efficiency. Unmanned aerial vehicles (UAVs), with their high mobility and favorable line-of-sight (LoS) links, can act as dynamic aerial transmitters and receivers in BackCom, thereby mitigating channel attenuation and improving both communication reliability and energy efficiency. To enhance the data collection efficiency of UAV-assisted BackCom systems under a limited mission duration, this paper proposes a joint optimization method for communication resource allocation and UAV trajectory design under task time constraints. Specifically, a mixed-integer non-convex optimization problem is formulated to maximize the number of devices served by the UAV within a given task duration. The original problem is then decomposed into two subproblems, namely communication resource allocation optimization and UAV trajectory optimization. An iterative algorithm based on Block Coordinate Descent (BCD) and Successive convex approximation (SCA) is developed to obtain an efficient solution. Simulation results demonstrate that the proposed method can effectively increase the number of served devices within the specified mission time limit.

Keywords: backscatter communication; UAV communication; resource allocation; trajectory design

Check for updates

Academic Editor: Hyokyung Bahn

Received: 18 March 2026
Revised: 17 April 2026
Accepted: 19 April 2026
Published: 23 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The Internet of Things (IoT) is profoundly transforming modern life through large-scale data transmission and processing among massive numbers of interconnected devices [1,2]. However, in many practical application scenarios, such as environmental monitoring in remote areas and large-scale agricultural data collection, IoT systems face several critical challenges, including low data transmission rates, difficulty in device maintenance, and limited energy supply. Consequently, developing IoT data acquisition systems that are low-power, long-lifetime, and maintenance-free has become an important research focus in the IoT field.

Wireless Sensor Networks (WSNs) are a typical data acquisition application scenario in the Internet of Things, where their nodes typically rely on limited battery power. Energy-Efficient Clustering is an effective method to improve node energy efficiency and extend network lifetime, while machine learning-based clustering, routing, and energy management approaches represent important directions in the design of wireless sensor networks. The authors in [3] proposed two machine learning-based improvements to the Distributed Energy-Efficient Clustering protocol, integrating the K-Nearest Neighbours and K-Means algorithms to dynamically optimize cluster head selection, thereby enhancing energy efficiency and network lifetime. The authors in [4] introduced a reinforcement learning (RL) technique based on State–Action–Reward–State–Action (SARSA) learning, which determines how many routing requests to accept based on energy consumption rate and residual energy, achieving a good trade-off between energy consumption and stability. The authors in [5] developed an Enhanced Energy Optimization Model for Industrial WSNs, utilize machine learning algorithms to process collected sensor data, and optimize the efficiency of the energy system. The author in [6] proposed a two-phase machine learning strategy to balance energy conservation and data transmission delay, significantly improving metrics such as average energy consumption, end-to-end delay, and network lifetime. Although much research has been dedicated to improving the operational efficiency of WSNs, communication energy consumption remains a critical factor limiting the battery life of sensor nodes.

Backscatter communication (BackCom), as an energy-efficient and low-cost communication paradigm, is particularly suitable for IoT devices with stringent energy and cost constraints [7–9]. This technology enables passive backscatter devices (BDs) to transmit information without requiring complex and power-hungry radio frequency (RF) transmitters. Instead, data transmission is achieved by modulating incident carrier signals or ambient RF signals [10–13]. In monostatic backscatter systems, the carrier emitter (CE) and receiver are typically integrated into the same reader, resulting in a compact system architecture and flexible deployment. Such systems can achieve reliable communication over moderate distances and have been widely applied in low-power IoT scenarios, including smart logistics, warehouse management, environmental monitoring, and healthcare applications.

However, ground-based monostatic backscatter systems suffer from severe performance degradation at longer distances due to the double attenuation effect in both the forward and backward channels, which significantly reduces the transmission efficiency of BackCom systems [14–16]. Moreover, in scenarios such as disaster rescue, remote-area monitoring, or temporary large-scale sensor network deployment, ground infrastructure may be unavailable or unable to provide sufficient coverage. These limitations further restrict the reliability and service coverage of conventional backscatter networks.

In recent years, unmanned aerial vehicles (UAVs) have been widely regarded as promising aerial mobile platforms for next-generation communication networks, owing to their high mobility, favorable line-of-sight (LoS) links with low path loss, low deployment cost, and flexible mission configuration. By integrating UAVs with backscatter technology, UAV-assisted BackCom systems can be established, where a UAV can act as a CE or receiver. By dynamically approaching passive IoT devices, UAVs can significantly improve the channel quality of backscatter links and enable on-demand expansion of network coverage. Existing studies have demonstrated the significant potential of UAV-assisted BackCom systems in improving system throughput, extending network coverage, and enhancing energy efficiency [17–19].

UAV-assisted BackCom systems have significant application value in a variety of time-sensitive IoT scenarios. For example, in environmental monitoring applications, a large number of low-power sensing devices are distributed over wide geographic areas,

and they are required to collect and upload data within a limited time so as to ensure the timeliness and integrity of monitoring information [17]. In disaster emergency communication scenarios, terrestrial communication infrastructure may be damaged, and UAVs can serve as temporary communication platforms to rapidly gather data from sensing devices deployed in affected areas within a constrained mission duration, thereby supporting disaster assessment and decision-making [20]. In such scenarios, maximizing information acquisition efficiency within a limited time window is of critical importance. Moreover, in applications such as smart agriculture and industrial IoT, data collection tasks often need to be completed within predefined periods to support real-time monitoring and control [21]. Since backscatter devices generally rely on ambient energy harvesting, their communication capabilities are inherently limited. Therefore, efficient resource scheduling and UAV trajectory design are essential for improving the system service capability within a finite time horizon [22].

Though UAV-assisted BackCom systems have been extensively studied in recent years, most existing works have primarily focused on optimizing system throughput or energy efficiency [23–25]. In [23], the authors consider a three-node BackCom system consisting of a UAV, a BD, and a receiver. In this system, the UAV transmits RF signals to the BD, and the BD conveys its data to the receiver through backscatter transmission. Two protocols, the transmit-backscatter (TB) protocol and transmit-backscatter-relay (TBR) protocol, are proposed for scenarios where the direct link is available and unavailable, respectively. By jointly optimizing the UAV trajectory, energy allocation, and reflection coefficients, the proposed schemes aim to maximize the overall system throughput. In [24], the authors investigate a UAV-based BackCom network (UBCN) consisting of multiple BDs, multiple ground CEs, and a UAV. The UAV flies within the target area to collect data from the BDs. By jointly optimizing three blocks of variables, including BD scheduling, CE transmission power, and UAV trajectory, the energy efficiency of the UBCN is maximized under several practical constraints, such as BDs' minimum throughput requirements, minimum harvested energy constraints, and the maximum UAV speed constraint. In [25], the authors focused on a communication network composed of a UAV equipped with a CE and a receiver, along with multiple mobile BDs. Based on the known positions of BDs, a UAV-assisted BackCom optimization model is developed. The UAV trajectory, transmission power, and BD scheduling are jointly optimized to maximize the system energy efficiency. However, these studies typically assume that the UAV-based BackCom system has sufficient time resources or ignore stringent task completion time constraints, which makes them difficult to apply directly to time-sensitive IoT scenarios.

Regarding time-constrained optimization problems, several existing studies have investigated task completion time or service efficiency in UAV-assisted data collection systems. The authors in [26] considered timely data collection in UAV-assisted IoT scenario, where the data form different ground devices have heterogeneous deadlines. They jointly optimize device scheduling and UAV trajectory to maximize the possible number of devices while guaranteeing the minimum uploaded per device. The authors in [27] considered a UAV-assisted data collection scenario and jointly optimized user scheduling and UAV trajectory to minimize the total time required to complete all data collection tasks. By analyzing the structural properties of the optimal solution, they proposed a successive-hover-fly (SHF) trajectory model and further developed a low-complexity iterative algorithm based on this model. In addition, the authors in [28] further examined time-constrained data collection in more complex communication environments. By introducing a reconfigurable intelligent surface (RIS), they jointly optimized the UAV trajectory, device scheduling, and RIS phase shifts to maximize the number of successfully served devices within a limited time window. Owing to the highly dynamic and non-convex nature of the problem, a hybrid solution

framework combining deep reinforcement learning (DRL) and Block Coordinate Descent (BCD) was adopted.

However, although the above studies have investigated time-constrained data collection from different perspectives, they are still based on traditional active communication paradigms and do not consider the application of BackCom in ultra-low-power IoT scenarios.

Motivated by the above observations, this paper investigates the problem of maximizing the number of served devices in a UAV-assisted BackCom system within a given mission time. The main contributions of this paper are summarized as follows.

- A UAV-assisted BackCom system model is established, and a service device number maximization problem is formulated. Specifically, this paper considers a system in which a UAV simultaneously acts as a CE emitter and data receiver to collect data from multiple BDs. On this basis, with the objective of maximizing the number of served BDs within a given mission duration, a corresponding joint optimization problem is formulated by jointly considering communication resource allocation and UAV trajectory design.
- The original problem is formulated as a mixed-integer non-convex optimization problem, and an efficient solution framework is developed by decomposing the joint optimization problem into two subproblems that are solved alternately. Due to the strong coupling between the device scheduling variables and the UAV trajectory variables, the formulated problem is inherently mixed-integer and non-convex, making it challenging to solve directly. To tackle this difficulty, the discrete variables are first relaxed, based on which an iterative algorithmic framework combining BCD and SCA is developed. Specifically, the original joint optimization problem is decomposed into a communication resource allocation subproblem and a UAV trajectory design subproblem. The SCA technique is employed to deal with the non-convex constraints involved in trajectory optimization, while the communication resource allocation variables and the UAV trajectory variables are updated alternately through an alternating optimization procedure.
- By simulation, the effectiveness and superiority of the proposed algorithm are validated. The results show that, compared with the benchmark schemes, the proposed joint optimization algorithm for communication resource allocation and UAV trajectory can significantly increase the number of devices served by the UAV-assisted BackCom system within a given task duration.

The remainder of this paper is organized as follows. Section 2 introduces the system model. Section 3 presents the proposed BCD and SCA-based joint optimization algorithm in detail. Section 4 provides the simulation results and corresponding analysis. Section 5 concludes the paper.

2. System Model

Consider a UAV-assisted BackCom scenario, where both the CE and the receiver are mounted on a single UAV, as illustrated in Figure 1. A total of I BDs are deployed on the ground in remote areas with limited infrastructure, where continuous and long-term power supply for IoT devices is difficult to guarantee. By adopting BackCom, these BDs can operate without batteries and rely solely on external RF signals for data transmission. This approach significantly reduces the hardware cost, lowers maintenance requirements, and prolongs the lifetime of BDs. The set of BDs is denoted by $B = \{BD_1, BD_2, \ldots, BD_I\}$, where BD_i represents the i-th BD. Each BD is associated with one sensor to collect environmental data. A UAV is dispatched to gather data from the BDs through BackCom and deliver the collected information to the control center before the mission deadline. Both the UAV and the BDs are equipped with a single antenna.

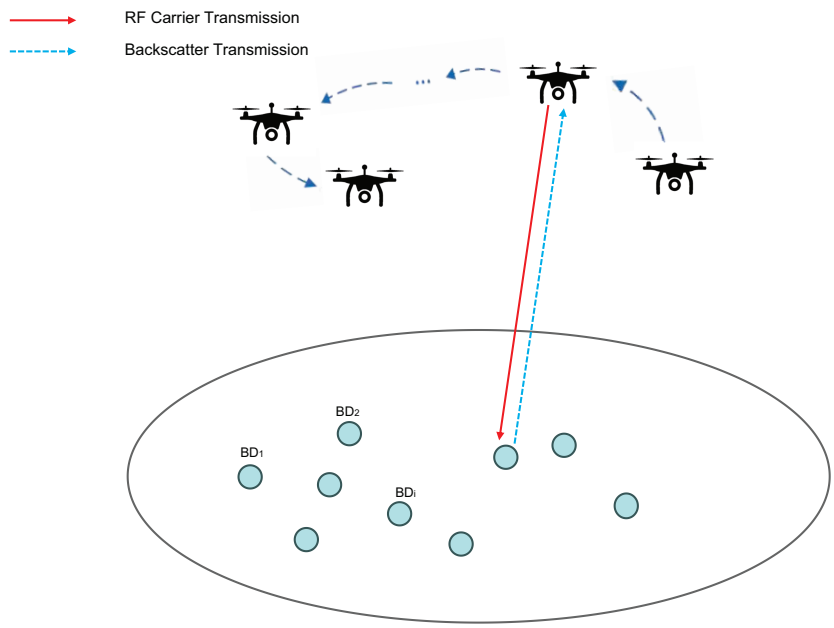


Figure 1. System model.

It is assumed that the locations of all BDs and the amount of data to be transmitted by each BD are fixed, and these parameters are known to the control center. The location of BD_i is denoted by (x_i, y_i), and the amount of data to be collected is denoted by C_i. Based on the deployment locations of all BDs, the corresponding data volumes, and the data collection task duration constraint, the control center pre-plans the trajectory before the UAV takes off.

The flight duration of each UAV mission, which corresponds to the task duration, is denoted by T. The task duration T is divided into N equal time slots, where the n-th slot is indexed by t_n, and the duration of each slot is denoted by δ_t. We assume the time-slot duration is sufficiently small, and thus the UAV movement within each time slot can be considered negligible compared with the distances between the UAV and BDs. Therefore, the channel gain remains constant within the same slot. The UAV flies at a constant altitude H above the ground. Its position at slot t_n can be represented by (x_n^U, y_n^U). The initial taking-off position of the UAV is denoted by (x_s, y_s), and the final landing position is denoted by (x_e, y_e).

The communication link from the CE on the UAV to BD_i, as well as the communication link from BD_i to the receiver, are modeled based on the free-space path-loss model. Therefore, the channel power gain between the CE and BD_i can be expressed as $B_i = \frac{B_0}{d_{i,n}^2}$, where B_0 denotes the channel power gain at a reference distance of 1 m [24]. Since the CE and the receiver are both mounted on the UAV, the uplink and downlink channels are treated as symmetric links, and the channel power gain between BD_i and the receiver on the UAV is also B_i.

3. Joint Resource Allocation and Trajectory Design for UAV-Assisted BackCom Systems Under Time Constraints

3.1. Optimization Problem Formulation

The objective of this paper is to maximize the total number of IoT devices served within the given mission time T, by jointly optimizing the communication resource allocation and the UAV flight trajectory.

BD_i is located on the ground, and its position is denoted by (x_i, y_i). The UAV flies at a fixed altitude H, and its two-dimension position at slot t_n is denoted by (x_n^U, y_n^U). Therefore, the distance between the UAV and BD_i at slot t_n can be expressed as follows.

$$d_{i,n} = \sqrt{(x_i - x_n^U)^2 + (y_i - y_n^U)^2 + H^2} \tag{1}$$

During flight, the maximum speed of the UAV is limited to V_{max}. Therefore, within a single slot, the distance traveled by the UAV must satisfy the following constraints.

$$\left(x_{n+1}^U - x_n^U\right)^2 + \left(y_{n+1}^U - y_n^U\right)^2 \leq (V_{\max}\delta_t)^2, \quad n = 1, 2, \ldots, N-1 \tag{2}$$

Assume that at slot t_n, the UAV flies over and communicates with BD_i. The signal transmission path is CE–BD–receiver, where both the CE and the receiver are mounted on the UAV. The CE transmits a carrier signal to BD_i with constant power p_0. Then, BD_i modulates the incident signal through backscatter and reflects it to the receiver located on the UAV.

Accordingly, the signal-to-noise ratio (SNR) of the signal received by the UAV from BD_i can be expressed as $\gamma_{i,n} = \frac{B_0^2 P_0}{d_{i,n}^4 \sigma^2}$. Here, σ^2 denotes the additive white Gaussian noise (AWGN) power at the receiver on the UAV.

Therefore, the amount of data received by the UAV from BD_i during slot t_n can be denoted by

$$Q_i^n = \delta_t \log_2(1 + \gamma_{i,n}) = \delta_t \log_2\left(1 + \frac{B_0^2 P_0}{d_{i,n}^4 \sigma^2}\right). \tag{3}$$

Since the UAV can communicate with only one BD in each time slot, a binary scheduling variable $b_i(n)$ is introduced to represent the communication relationship between the UAV and BD_i during slot t_n. Specifically, when BD_i communicates with the UAV, $b_i(n) = 1$, otherwise, $b_i(n) = 0$. Furthermore, since the UAV can communicate with at most one BD during each slot t_n, the constraint $0 \leq \sum_{i=1}^{I} b_i(n) \leq 1$ holds.

The total amount of data transmitted by BD_i in the uplink over all slots can be represented as

$$Q_i = \sum_{n=1}^{N} b_i(n)\, \delta_t \log_2\left(1 + \frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right). \tag{4}$$

In this paper, the binary data integrity variable k_i is introduced to indicate whether the data of BD_i has been completely uploaded. Specifically, when $k_i = 0$, it indicates that the data of BD_i has not been fully transmitted; when $k_i = 1$, it indicates that the data of BD_i has been successfully transmitted, i.e.,

$$k_i = \begin{cases} 0, & Q_i < C_i \\ 1, & Q_i \geq C_i \end{cases}. \tag{5}$$

The horizontal coordinate vector of the UAV is represented as an N-dimensional real vector $\mathbf{x}^u = [x_1^u, x_2^u, \ldots, x_N^u]^T$. The vertical coordinate vector is represented as an N-dimensional real vector $\mathbf{y}^u = [y_1^u, y_2^u, \ldots, y_N^u]^T$. The state vector indicating whether the data of each BD is successfully transmitted is denoted as $\mathbf{k} = [k_1, k_2, \ldots, k_I]^T$.

The communication resource allocation vector is defined as $\mathbf{b}_i = [b_i(1), b_i(2), \ldots, b_i(N)]^T$ for BD_i. The I communication resource allocation vectors can be constructed as an $N \times I$ resource allocation matrix $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_I]$. The element in the n-th row and i-th column of matrix $\mathbf{B}$ is $b_i(n)$.

The optimization problem can be formulated as $P1$, whose objective is to maximize the number of served BDs within one specific task duration constraint.

$$
\begin{aligned}
\text{P1}: \quad & \max_{\mathbf{x}^u,\mathbf{y}^u,\mathbf{k},\mathbf{b}_i} \sum_{i\in I} k_i \\
\text{s.t.} \quad & (x^u_{n+1}-x^u_n)^2+(y^u_{n+1}-y^u_n)^2 \le (V_{max}\delta_t)^2, \quad n=1,2,\ldots,N-1 \qquad (1a)\\
& 0 \le \textstyle\sum_{i=1}^{I} b_i(n) \le 1 \qquad (1b)\\
& \textstyle\sum_{n=1}^{N} b_i(n)\delta_t \log_2\left(1+\frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right) \ge k_i C_i, \quad i=1,2,\ldots,I \qquad (1c)\\
& k_i \in \{0,1\}, \quad i \in I \qquad (1d)\\
& b_i(n) \in \{0,1\}, \quad i \in I,\ n=1,2,\ldots,N \qquad (1e)
\end{aligned}
$$

Constraint (1a) represents the UAV traveling distance constraint within a single time slot, i.e., the maximum flying distance constraint. Constraint (1b) indicates that the UAV can communicate with at most one BD within each time slot. Constraint (1c) indicates that, over all N slots, if the data transmission of BD_i is successful, the total amount of data transmitted to the UAV should be no less than the total data stored in BD_i. Constraint (1d) represents the data integrity status of BD_i. Constraint (1e) represents the communication state between BD_i and the UAV in the n-th slot.

3.2. Joint Optimization of Resource Allocation and Trajectory

For the optimization problem P1 described above, it is evident that the problem is a mixed-integer non-convex optimization problem (MINLP), and is therefore difficult to solve. On the one hand, the variables k_i and $b_i(n)$ are discrete, which renders the problem combinatorial in nature. On the other hand, the constraints involve non-convex functions introduced by the UAV trajectory, which further increases the difficulty of problem solving.

To address this issue, k_i is first relaxed to $0 \le k_i \le 1$, and $b_i(n)$ is relaxed to $0 \le b_i(n) \le 1$. Accordingly, P1 can be reformulated as P1.1.

$$
\begin{aligned}
& \max_{\mathbf{x}^u,\mathbf{y}^u,\mathbf{k},\mathbf{b}_i} \sum_{i\in I} k_i \qquad P1.1\\
\text{s.t.} \quad & (x^u_{n+1}-x^u_n)^2+(y^u_{n+1}-y^u_n)^2 \le (V_{max}\delta_t)^2, \quad n=1,2,\ldots,N-1 \qquad (1.1a)\\
& 0 \le \textstyle\sum_{i=1}^{I} b_i(n) \le 1 \qquad (1.1b)\\
& \textstyle\sum_{n=1}^{N} b_i(n)\delta_t \log_2\left(1+\frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right) \ge k_i C_i, \quad i=1,2,\ldots,I \qquad (1.1c)\\
& 0 \le k_i \le 1, \quad i \in I \qquad (1.1d)\\
& 0 \le b_i(n) \le 1, \quad i \in I,\ n=1,2,\ldots,N \qquad (1.1e)
\end{aligned}
$$

In problem $P1.1$, it is obvious that the objective function as well as constraints (1.1b), (1.1d), and (1.1e) are linear. In constraint (1.1a), we define the following function

$$f_n(x^n,y^n,x^{n+1},y^{n+1}) = (x^{n+1}-x^n)^2+(y^{n+1}-y^n)^2-(V_{max}\delta_t)^2. \qquad (6)$$

The computed Hessian matrix of $f_n(x^n,y^n,x^{n+1},y^{n+1})$ is positive semidefinite (PSD), indicating that f_n is a convex function. Therefore, constraint (1.1a) is a convex constraint.

In constraint (1.1c), the left-hand side is the product of a function of the resource allocation vector $\mathbf{b}_i$ and a function of the coordinate vector (x^u,y^u). Therefore, this constraint is non-convex.

To improve the tractability of the problem, this paper adopts an optimization framework based on the BCD method [29], in which the original problem is decomposed into two subproblems, namely, communication resource allocation and UAV trajectory optimization, which are solved through an alternating optimization procedure.

In particular, given the UAV trajectory, the resource allocation subproblem can be transformed into a convex optimization problem. In contrast, due to its non-convex nature, the trajectory optimization subproblem will be addressed using the SCA method [30]. This approach is capable of obtaining a high-quality locally optimal solution while maintaining manageable computational complexity.

3.2.1. Communication Resource Allocation Optimization

In Subproblem 1, given the UAV flight trajectory $\mathbf{x}^u$ and $\mathbf{y}^u$, the communication resource allocation matrix $\mathbf{B}$ and the transmission result vector $\mathbf{k}$ are optimized. In this case, constraint (1.1a) always holds, while constraints (1.1b), (1.1d), and (1.1e) are linear. For constraint (1.1c), define

$$R_{i,n} = \log_2\left(1 + \frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right). \tag{7}$$

Since $d_{i,n}$ is a constant under the given UAV trajectory, $R_{i,n}$ is also a constant. Therefore, constraint (1.1c) can be rewritten as the following linear constraint.

$$\sum_{n=1}^{N} b_i(n)\delta_t R_{i,n} \geq k_i C_i \tag{8}$$

Accordingly, problem $P1.1$ can be reformulated as

$$\begin{aligned}
&\max_{\mathbf{k},\mathbf{b}_i} \sum_{i\in I} k_i && (P2)\\
\text{s.t.}\quad & 0 \leq \textstyle\sum_{i=1}^{I} b_i(n) \leq 1 && (2a)\\
& \textstyle\sum_{n=1}^{N} b_i(n)\delta_t R_{i,n} \geq k_i C_i, \quad i = 1,2,\ldots,I && (2b)\\
& 0 \leq k_i \leq 1, \quad i \in I && (2c)\\
& 0 \leq b_i(n) \leq 1, \quad i \in I,\ n = 1,2,\ldots,N && (2d)
\end{aligned}$$

The above problem is a linear optimization problem and can be directly solved by using CVX.

3.2.2. UAV Trajectory Optimization

In Subproblem 2, given the communication resource allocation vector $\mathbf{b}_i$, the UAV flight trajectories $\mathbf{x}^u$ and $\mathbf{y}^u$, as well as the transmission result vector $\mathbf{k}$, are optimized. In this case, $b_i(n)$ is a known constant, and constraints (1.1b) and (1.1e) can be omitted. Accordingly, problem $P1.1$ can be simplified as

$$\begin{aligned}
&\max_{\mathbf{x}^u,\mathbf{y}^u,\mathbf{k}} \sum_{i\in I} k_i && (P3)\\
\text{s.t.}\quad & (x_{n+1}^u - x_n^u)^2 + (y_{n+1}^u - y_n^u)^2 \leq (V_{max}\delta_t)^2, \quad n = 1,2,\ldots,N-1 && (3a)\\
& \textstyle\sum_{n=1}^{N} b_i(n)\delta_t \log_2\left(1 + \frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right) \geq k_i C_i, \quad i = 1,2,\ldots,I && (3b)\\
& 0 \leq k_i \leq 1, \quad i \in I && (3c)
\end{aligned}$$

In problem $P3$, the objective function and constraint (3c) are linear. Constraint (3a) is a convex constraint. In constraint (3b), $b_i(n)$ and δ_t are known constants. Therefore, we only need to examine the concavity of function $R_{i,n}$. Since B_0, σ^2, and P_0 are constants, let $a = \frac{B_0^2 P_0}{\sigma^2}$, substitute $d_{i,n} = \sqrt{(x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2}$ in $R_{i,n}$, and we obtain

$$R_{i,n}(x_n^u, y_n^u) = \log_2\left(1 + \frac{a}{[(x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2]^2}\right). \tag{9}$$

The Hessian matrix of $R_{i,n}(x_n^u, y_n^u)$ is neither positive semidefinite nor negative semidefinite. Therefore, the function is neither convex nor concave.

To obtain a feasible locally optimal solution to non-convex problem $P3$, the SCA method is adopted. SCA is an iterative method for solving non-convex optimization problems. In each iteration, a convex approximation function is constructed based on the current iteration point to replace the original non-convex function. By solving the resulting convex problem, the next iteration point is determined, and thus a locally optimal solution of the original problem can be gradually approached.

First, define the slack variable vector $\mathbf{u}_i = [u_i(1), u_i(2), \ldots, u_i(N)]^T$. The I slack variable vectors form an $N \times I$ slack variable matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_I]$. The element in the n-th row and i-th column of the matrix is denoted by $\mu_{i,n}$, which satisfies

$$\mu_{i,n} \geq (x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2, \quad i = 1, 2, \ldots, I, \ n = 1, 2, \ldots, N. \tag{10}$$

After introducing the slack variable, problem $P3$ can be rewritten as

$$\max_{\mathbf{x}^u, \mathbf{y}^u, \mathbf{k}, \mathbf{U}} \sum_{i \in I} k_i \qquad (P3.1)$$

$$\text{s.t.} \quad (x_{n+1}^u - x_n^u)^2 + (y_{n+1}^u - y_n^u)^2 \leq (V_{max}\delta_t)^2, \quad n = 1, 2, \ldots, N-1 \qquad (3.1a)$$

$$k_i C_i - \sum_{n=1}^{N} b_i(n)\delta_t \log_2\left(1 + \frac{a}{\mu_{i,n}^2}\right) \leq 0, \quad i = 1, 2, \ldots, I \qquad (3.1b)$$

$$(x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2 - \mu_{i,n} \leq 0, \quad i = 1, 2, \ldots, I, \ n = 1, 2, \ldots, N \qquad (3.1c)$$

$$0 \leq k_i \leq 1, \quad i \in I \qquad (3.1d)$$

After introducing the slack variables, the additional constraint (3.1c) remains convex. The optimal solution of $P3.1$ satisfies

$$\mu_{i,n} = (x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2, \tag{11}$$

which implies that the optimal solutions of $P3$ and $P3.1$ are identical. Since function $R_{i,n}(\mu_{i,n}) = \log_2\left(1 + \frac{a}{\mu_{i,n}^2}\right)$ is convex, constraint (3.1b) is still non-convex. In this paper, the convex function $R_{i,n}(\mu_{i,n})$ is approximated by its first-order Taylor expansion. Specifically, a linear approximation is constructed at feasible point $(\mu_{i,n}^0, R_{i,n}(\mu_{i,n}^0))$. The linear approximation function of $R_{i,n}(\mu_{i,n})$ at $(\mu_{i,n}^0, R_{i,n}(\mu_{i,n}^0))$ is denoted as $\tilde{R}_{i,n}(\mu_{i,n} \mid \mu_{i,n}^0)$, expressed as

$$\begin{aligned} \tilde{R}_{i,n}(\mu_{i,n} \mid \mu_{i,n}^0) &\triangleq R_{i,n}(\mu_{i,n}^0) + R'_{i,n}(\mu_{i,n}^0)(\mu_{i,n} - \mu_{i,n}^0) \\ &= \log_2\left(1 + \frac{a}{(\mu_{i,n}^0)^2}\right) - \frac{2a(\mu_{i,n} - \mu_{i,n}^0)}{\ln 2\, \mu_{i,n}^0\left((\mu_{i,n}^0)^2 + a\right)}. \end{aligned} \tag{12}$$

Thus, problem $P3.1$ can be further reformulated as follows.

$$\max_{\mathbf{x}^u, \mathbf{y}^u, \mathbf{k}, \mathbf{U}} \sum_{i \in I} k_i \qquad (P3.2)$$

$$\text{s.t.} \quad (x_{n+1}^u - x_n^u)^2 + (y_{n+1}^u - y_n^u)^2 \leq (V_{max}\delta_t)^2, \quad n = 1, 2, \ldots, N-1 \qquad (3.2a)$$

$$k_i C_i - \sum_{n=1}^{N} b_i(n)\delta_t \tilde{R}_{i,n}(\mu_{i,n}) \leq 0, \quad i = 1, 2, \ldots, I \qquad (3.2b)$$

$$(x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2 - \mu_{i,n} \leq 0, \quad i = 1, 2, \ldots, I, \ n = 1, 2, \ldots, N \qquad (3.2c)$$

$$0 \leq k_i \leq 1, \quad i \in I \qquad (3.2d)$$

Thus, the convex problem $P3.2$ can be directly solved using CVX. In each iteration of the SCA algorithm, the value of $\mu_{i,n}^0$ is updated according to the solution $\mathbf{x}^u$ and $\mathbf{y}^u$ obtained in the previous iteration, i.e.,

$$\mu_{i,n}^0 = (x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2, \quad i = 1, 2, \ldots, I, \ n = 1, 2, \ldots, N. \tag{13}$$

Then, by substituting the updated $\mu_{i,n}^0$ into $P3.2$, the values of $\mathbf{x}^u$ and $\mathbf{y}^u$ can be updated. In the next iteration, $\mu_{i,n}^0$ is determined again, and $\mathbf{x}^u$ and $\mathbf{y}^u$ are re-optimized accordingly. The pseudocode of the trajectory optimization scheme based on SCA is presented in Algorithm 1. The obtained maximum number of served devices within the given time is denoted as $g(\mathbf{k})$, i.e.,

$$g(\mathbf{k}) = \max_{\mathbf{x}^u, \mathbf{y}^u, \mathbf{k}, \mathbf{U}} \sum_{i \in I} k_i. \tag{14}$$

Algorithm 1: SCA-based Trajectory Optimization Scheme

Input: Initial UAV trajectory x_0^u, y_0^u, communication resource allocation b_i, initial iteration $j = 0$, convergence threshold ε_1.
Output: Optimized UAV trajectory x^u, y^u and the maximum number of served devices $g(\mathbf{k})$.
Initialize feasible point $\mu_{i,n}^0 = [(x_i - x_n^u)^2 + (y_i - y_n^u)^2 + H^2]$.
repeat
  Update feasible point $(\mu_{i,n}^0, R_{i,n}(\mu_{i,n}^0))$.
  Construct convex approximation function $\tilde{R}_{i,n}(\mu_{i,n} \mid \mu_{i,n}^0)$.
  Solve convex approximation problem (P3.2) to update x^u, y^u.
  $j = j + 1$.
until $|g^j - g^{j-1}| < \varepsilon_1$.

Algorithm 2 presents the pseudocode of the joint optimization scheme for communication resource allocation and UAV trajectory under the task duration constraint. In each iteration, the maximum number of served devices, denoted by $g(\mathbf{k})$, can be updated. When the difference between the results of two consecutive iterations is smaller than the convergence threshold ε_2, the iteration process terminates.

Algorithm 2: Joint Resource Allocation and Trajectory Design Optimization Scheme

Input: Number of BDs I, number of time slots N, BD locations (x_i, y_i), data size C_i, slot duration δ_t, CE power p_0, UAV altitude H, channel gain B_0, convergence threshold ε_2.
Output: UAV trajectory $\mathbf{x}^u, \mathbf{y}^u$, communication resource allocation b_i, and the maximum number of served devices $g(\mathbf{k})$.
Initialize optimization variables x^u, y^u, b_i, outer iteration index $l = 0$.
repeat
  Given x^u, y^u, solve problem (P2) to update b_i.
  Given b_i, apply Algorithm 1 to solve problem (P3) and update $x^u, y^u, g(\mathbf{k})$.
  $l = l + 1$.
until $|g(\mathbf{k}, l) - g(\mathbf{k}, l-1)| < \varepsilon_2$.

3.3. Computational Complexity Analysis

The proposed joint optimization algorithm is developed based on a framework combing BCD and SCA, and the original problem is solved by alternately optimizing the com-

munication resource allocation and the UAV trajectory. The overall algorithm consists of an outer BCD iteration loop and an inner SCA iteration loop, and its computational complexity mainly arises from the solution procedures of these two subproblems.

Given the UAV trajectory, the resource allocation subproblem P2 can be transformed into a linear optimization problem. The optimization variables in this subproblem mainly include the device scheduling variables k_i and the resource allocation variables $b_i(n)$, whose problem size is on the order of $O(IN)$, where I denotes the number of devices and N denotes the number of time slots. The number of constraints is of the same order as the number of variables, i.e., approximately $O(IN)$. Therefore, this subproblem can be efficiently solved by standard convex optimization tools, with a computational complexity of approximately $O\big((IN)^3\big)$.

Given the communication resource allocation, the trajectory optimization subproblem P3 is non-convex. In this paper, the SCA method is employed to transform it into a sequence of convex optimization problems, namely, P3.2, which are solved iteratively. In each SCA iteration, the optimization variables include the UAV trajectory variables as well as the introduced slack variables, and the overall problem size can be approximated as $O(IN)$. Therefore, the computational complexity of solving each convex subproblem is on the order of $O\big((IN)^3\big)$. If the number of inner SCA iterations is denoted by J, then the computational complexity of the trajectory optimization part is $O\big(J(IN)^3\big)$.

If the number of outer BCD iterations is denoted by L, then the overall computational complexity of the proposed algorithm can be expressed as $O\big(L(J+1)(IN)^3\big)$. Although the proposed method requires solving the subproblems iteratively over multiple rounds, its computational complexity remains within an acceptable range. Moreover, the algorithm is more suitable for offline optimization or quasi-real-time task-planning scenarios, such as periodic data collection or application environments where the device distribution is known in advance. In such scenarios, the real-time computational burden can be alleviated through offline pre-computation or stage-wise updates, thereby ensuring the practical feasibility of the proposed method.

3.4. Reconstruction of Binary Resource Allocation and Number of Served BDs

When transforming the original problem $P1$ into $P1.1$, the binary communication resource allocation variable $b_i(n)$ and the data integrity variable k_i, which indicates whether the data of BD is successfully transmitted, are relaxed into continuous variables within the interval $[0, 1]$. Therefore, in the solutions obtained from the alternating optimization of the two subproblems, if both the communication resource allocation variable $b_i(n)$ and the data integrity variable k_i are binary, the relaxation is tight, and the obtained solution is also a feasible solution to $P1$. However, when the value of $b_i(n)$ is fractional, the result of the alternating optimization cannot be directly used as a feasible solution to $P1$, and reconstruction of $b_i(n)$ and k_i is required.

Similar to the reconstruction process described in [31], each time slot in this paper is further divided into M sub-slots, resulting in a total number of sub-slots $N' = NM$. The sub-slot is denoted by $t_{n,m}$, and the binary communication resource allocation variable is defined as $\bar{b}_i(n, m) \in \{0, 1\}$. During slot t_n, the number of sub-slots allocated by the UAV to BD_i is denoted as

$$N_{BD}(i, n) = \lfloor M b_i(n) \rceil, \tag{15}$$

where $\lfloor x \rceil$ represents the nearest integer to x. As M increases, $N_{BD}(i, n)$ approaches an integer, and thus a binary solution can be obtained.

For example, within slot t_n, suppose $b_1(n) = 0.59$ and $b_2(n) = 0.41$. If slot t_n is divided into 10 sub-slots, i.e., $M = 10$, then we have $N_{BD}(1, n) = \lfloor M b_1(n) \rceil = 6$ and $N_{BD}(2, n) = \lfloor M b_2(n) \rceil = 4$. In this case, the UAV allocates 6 sub-slots to BD_1 and 4 sub-slots to BD_2.

When $M = 100$, we have $N_{BD}(1,n) = \lfloor Mb_1(n) \rceil = 59$ and $N_{BD}(2,n) = \lfloor Mb_2(n) \rceil = 41$. Thus, the UAV allocates 59 sub-slots to BD_1 and 41 sub-slots to BD_2.

After the reconstruction, the total amount of transmitted data for each user is recalculated using the new binary scheduling variable $\bar{b}_i(n,m)$, which can be expressed as

$$Q_i = \sum_{n=1}^{N} \sum_{m=1}^{M} \bar{b}_i(n,m) \cdot \frac{\delta_t}{M} \log_2\left(1 + \frac{B_0^2 P_0}{\sigma^2 d_{i,n}^4}\right). \tag{16}$$

Here, the duration of each sub-slot is δ_t / M, and the channel gain $\frac{B_0}{d_{i,n}^2}$ depends on the UAV position within the original time slot t_n. The transmission status of BD_i is then re-evaluated. The re-evaluated transmission status of BD_i is denoted by $\bar{k}_i$. If all data of BD_i are successfully transmitted, then $\bar{k}_i = 1$; otherwise, $\bar{k}_i = 0$.

Therefore, according to the reconstructed $\bar{k}_i$, the final value of $g(\mathbf{k})$ can be obtained as

$$g(\mathbf{k}) = \sum_{i \in I} \bar{k}_i. \tag{17}$$

4. Simulation Results

In this section, numerical simulation results are presented to evaluate the performance of the proposed algorithm. A ground area of 20×20 m^2 is considered, where I BDs are randomly deployed. The CE is mounted on the UAV with transmit power $p_0 = 6$ W. When the distance between the BD and the CE is 1 m, the channel power gain is 1×10^{-5}. For the UAV, it always flies with the maximum speed $V_{max} = 10$ m/s at a fixed altitude $H = 5$ m. The initial and final positions of the UAV are given by $q_s = (0,0)$ and $q_e = (20,20)$, respectively. The noise power at the receiver is $\sigma^2 = -144$ dBm. The duration of each time slot is set to $\delta_t = 0.3$.

For comparison, two greedy Traveling Salesman Problem (TSP) algorithm are considered as baseline schemes. At each step, the unvisited BD with the highest cost–performance ratio is selected. Within the given time constraint, priority is given to BDs with larger service demand and shorter flight distance.

4.1. Performance of the Proposed Joint Resource Allocation and Trajectory Design Optimization Scheme

4.1.1. Heterogeneous BD Data Sizes

To evaluate the performance of the proposed algorithm, two representative scenarios with the number of BDs set to $I = 6$ and $I = 10$ are considered. In the heterogeneous data size case, the amount of data to be uploaded by each even-indexed BD is assumed to be twice that of each odd-indexed BD.

Figures 2 and 3 illustrate the convergence of the objective value with the iteration number under different parameter settings. For different numbers of time slots N, the proposed BCD-SCA algorithm converges within eight iterations, demonstrating favorable convergence performance. Moreover, all curves exhibit a pronounced increasing trend during the first few iterations, indicating that the proposed algorithm is able to achieve significant performance improvement at the initial stage of the optimization process.

The convergence curves corresponding to different values of N exhibit similar variation trends, and the numbers of iterations required for convergence increase with the number of BDs I. As N increases, the final converged objective value gradually improves. This is because a larger number of time slots provides the UAV with greater trajectory flexibility, thereby enabling it to serve more BDs and enhancing the overall system performance.

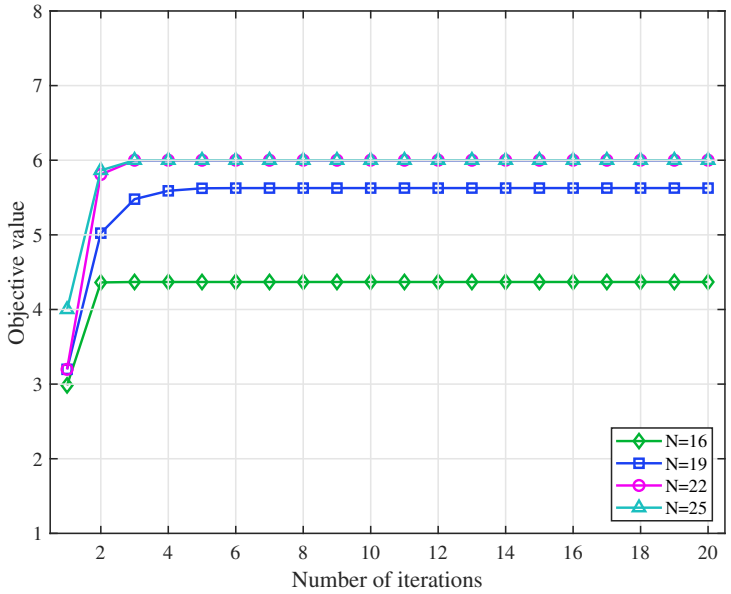


Figure 2. Convergence Behavior of the Proposed Algorithm for $I = 6$ with Heterogeneous BD Data Sizes.

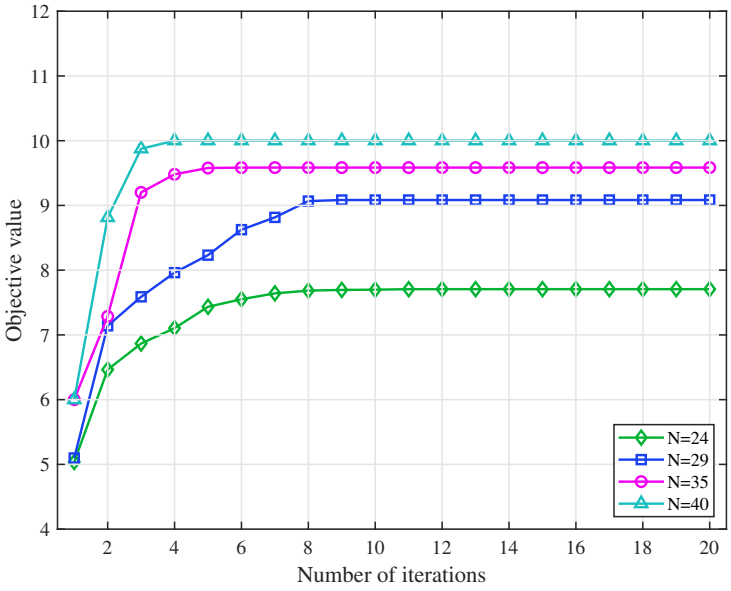
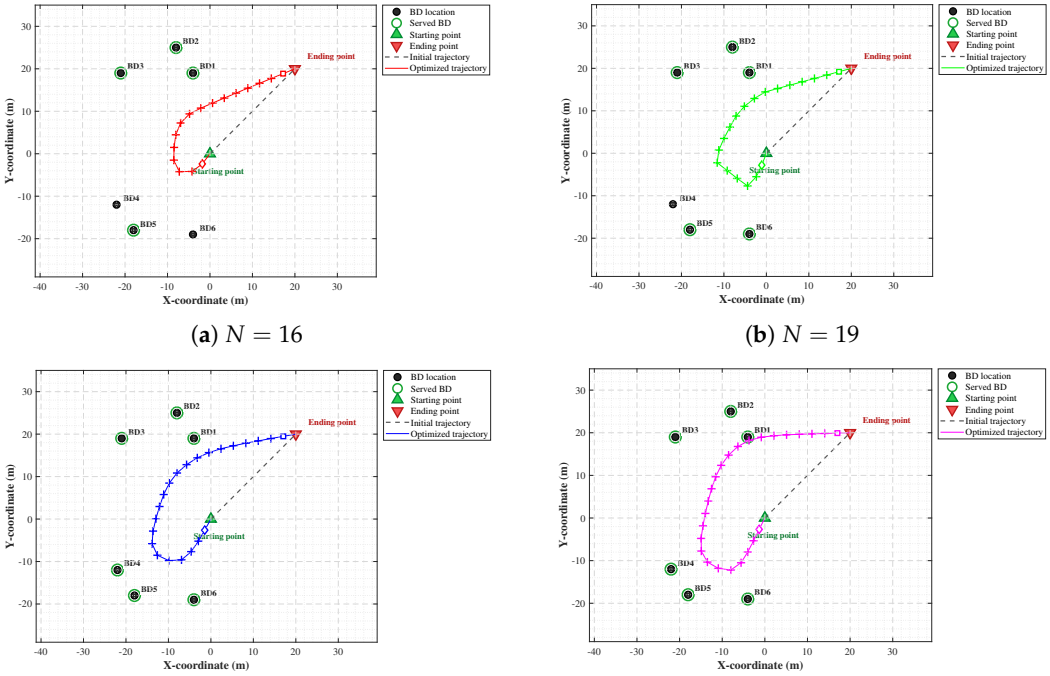


Figure 3. Convergence Behavior of the Proposed Algorithm for $I = 10$ with Heterogeneous BD Data Sizes.

Figure 4a–d depict the optimal UAV trajectories under different numbers of time slots when the number of BDs is $I = 6$, where the green circles denote the successfully served BDs. It can be observed that, when the number of time slots is small and insufficient to support the completion of data transmission for all BDs, the UAV tends to give priority to those BDs that are either closer to the initial and final locations or require a smaller amount of data to be transmitted. For example, when $N = 16$, the UAV completes data collection for BD_1, BD_2, and BD_3, which are relatively closer to both the starting point and the destination. As for the farther devices BD_4, BD_5, and BD_6, only those with smaller data demands are served successfully. This is because the UAV must return to the final location within the prescribed mission time, and its flight trajectory therefore needs to account for both the initial and terminal positions. Since the optimization objective is to maximize the number of served devices, when the distances are comparable, the UAV preferentially serves BDs with smaller data loads. When $N = 19$, only BD_4 fails to complete

data transmission. At the beginning of the flight, the UAV tends to move toward BD_5 and BD_6, since under a less stringent time budget, approaching the served BDs as much as possible can improve the transmission rate. Meanwhile, under the limited total mission duration, the UAV gives up serving BD_4, which is the farthest and has the largest data demand, so as to maximize the number of completed tasks within the given time. When $N = 22$ and $N = 25$, the available time becomes sufficient for all BDs to complete data transmission. In this case, the UAV initially flies farther toward BD_4, which is both the farthest device and the one with the largest data demand, and then proceeds toward the destination.



(**a**) $N = 16$ (**b**) $N = 19$

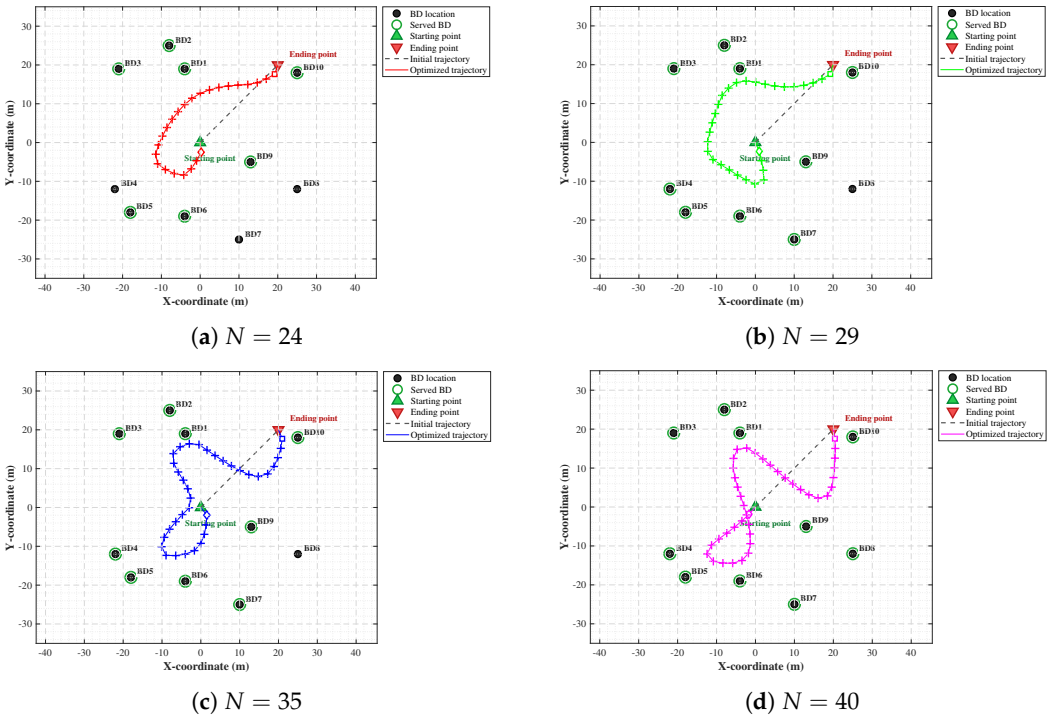
(**c**) $N = 22$ (**d**) $N = 25$

Figure 4. Optimized UAV Trajectories with Heterogeneous BD Data Sizes and a Straight-Line Initial Path when $I = 6$, (**a**) $N = 16$, (**b**) $N = 19$, (**c**) $N = 22$, (**d**) $N = 25$.

Figure 5a–d illustrate the optimal UAV trajectories under different numbers of time slots when the number of BDs is $I = 10$, where the green circles denote the successfully served BDs. When $N = 24$, seven BDs can be successfully served, while the BDs relatively far from both the initial and final locations or requiring larger data size upload remain unserved. When $N = 29$ and $N = 35$, only BD_8 fails to complete data transmission. This is because BD_8 has a relatively large data demand and is located farther away from the other BDs, making data collection for BD_8 less cost-effective under a limited time budget. When $N = 40$, the available time becomes sufficient, and all BDs are able to complete data transmissions.

The optimized trajectories exhibit a clear detour characteristic, where the UAV tends to first approach certain key BD regions and then performs localized adjustments nearby so as to improve the communication link quality. This trajectory pattern indicates that the proposed algorithm is capable of adaptively balancing the tradeoff between flight distance and communication performance, rather than simply following the shortest flight path. In addition, as the number of time slots N increases, the UAV trajectory gradually becomes smoother and more flexible, enabling it to cover more BD regions and thereby enhance the overall system performance. This is because a larger number of time slots provides

greater freedom for trajectory optimization, allowing the UAV to adjust its flight path more precisely to accommodate different spatial distributions of the BDs.



(**a**) $N = 24$ (**b**) $N = 29$

(**c**) $N = 35$ (**d**) $N = 40$

Figure 5. Optimized UAV Trajectories with Heterogeneous BD Data Sizes and a Straight-Line Initial Path when $I = 10$, (**a**) $N = 24$, (**b**) $N = 29$, (**c**) $N = 35$, (**d**) $N = 40$.

Figures 6 and 7 compare the relaxed solution and the reconstructed solution for the cases of $I = 6$ and $I = 10$, respectively. The gap between the reconstructed solution and the relaxed solution is relatively small. The discrepancy between the two mainly arises because only partial data is transmitted for some devices and the entire data collection task is not fully completed.

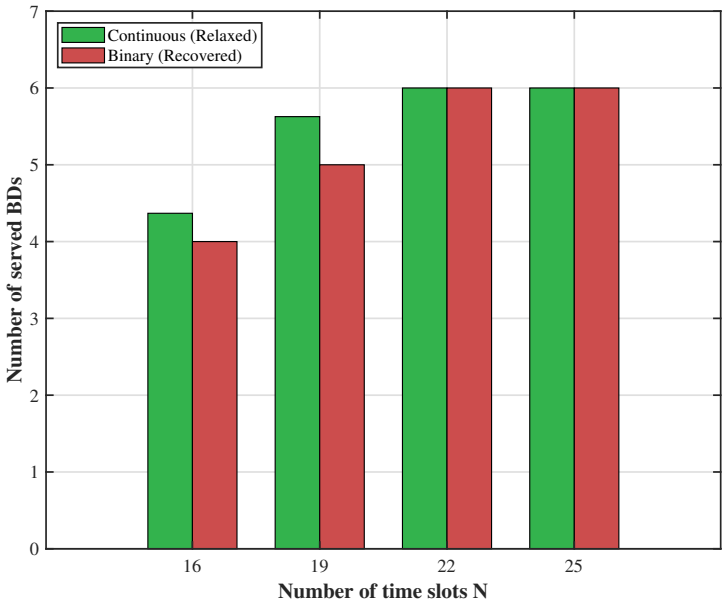


Figure 6. Comparison Between Relaxed and Reconstructed Solutions for $I = 6$ with Heterogeneous BD Data Sizes.

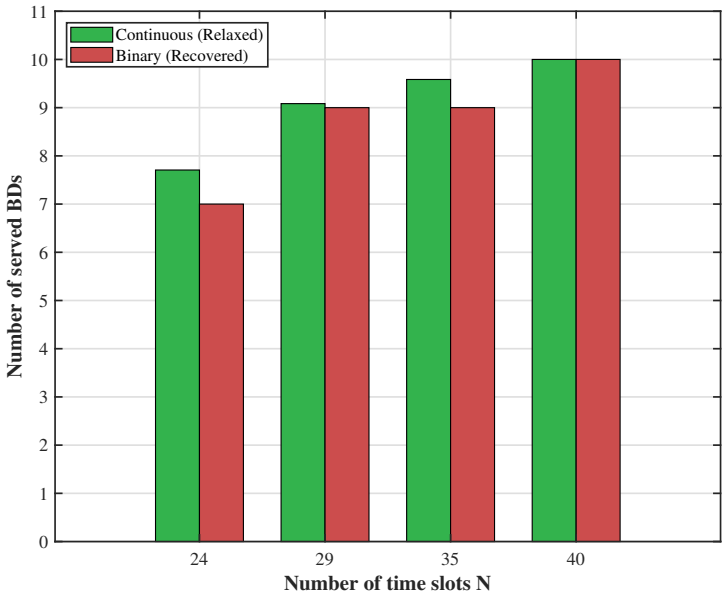


Figure 7. Comparison Between Relaxed and Reconstructed Solutions for $I = 10$ with Heterogeneous BD Data Sizes.

4.1.2. Homogeneous BD Data Sizes

To evaluate the impact of BD data sizes on the optimization results, this subsection considers the heterogeneous data size case. The total amount of data and the initial trajectory are kept the same as those in Section 4.1.1. Two representative scenarios with $I = 6$ and $I = 10$ are considered. The evolution of the optimization results with respect to the iteration number are shown in Figures 8 and 9. It can be observed that, under different settings of the number of time slots N, the proposed BCD-SCA algorithm also converges within limited iterations, demonstrating similar favorable convergence behavior. Furthermore, compared with heterogeneous data size case in Section 4.1.1, the required numbers of iterations differ only slightly, which further confirms the robustness of the proposed algorithm.

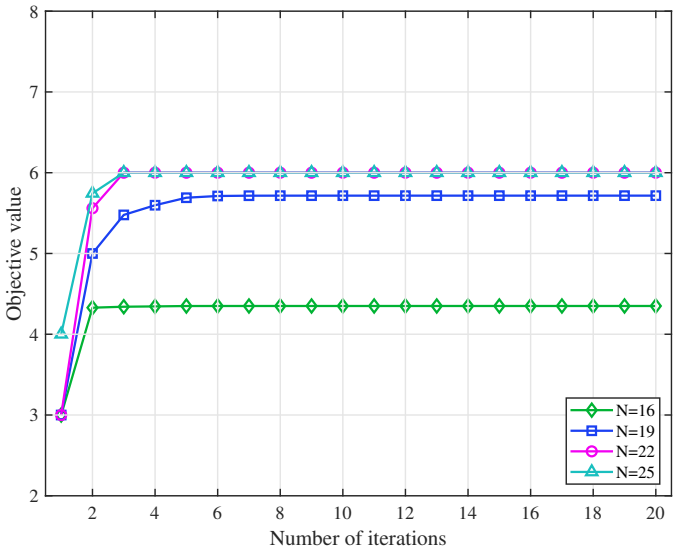


Figure 8. Convergence Behavior of the Proposed Algorithm for $I = 6$ with Homogeneous BD Data Sizes.

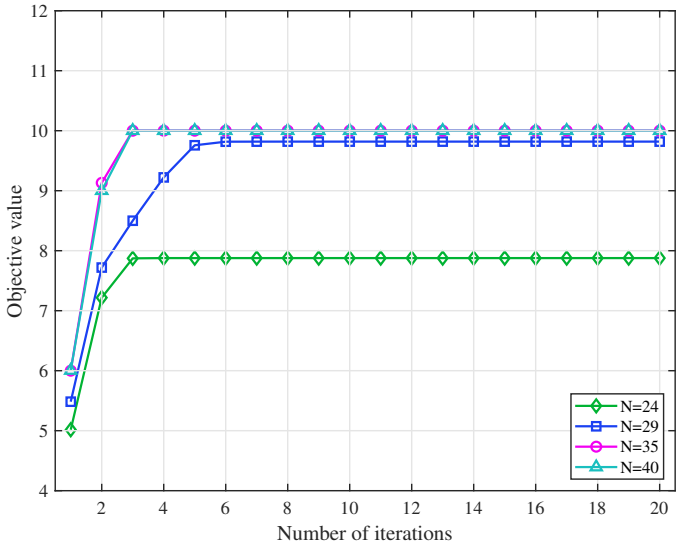
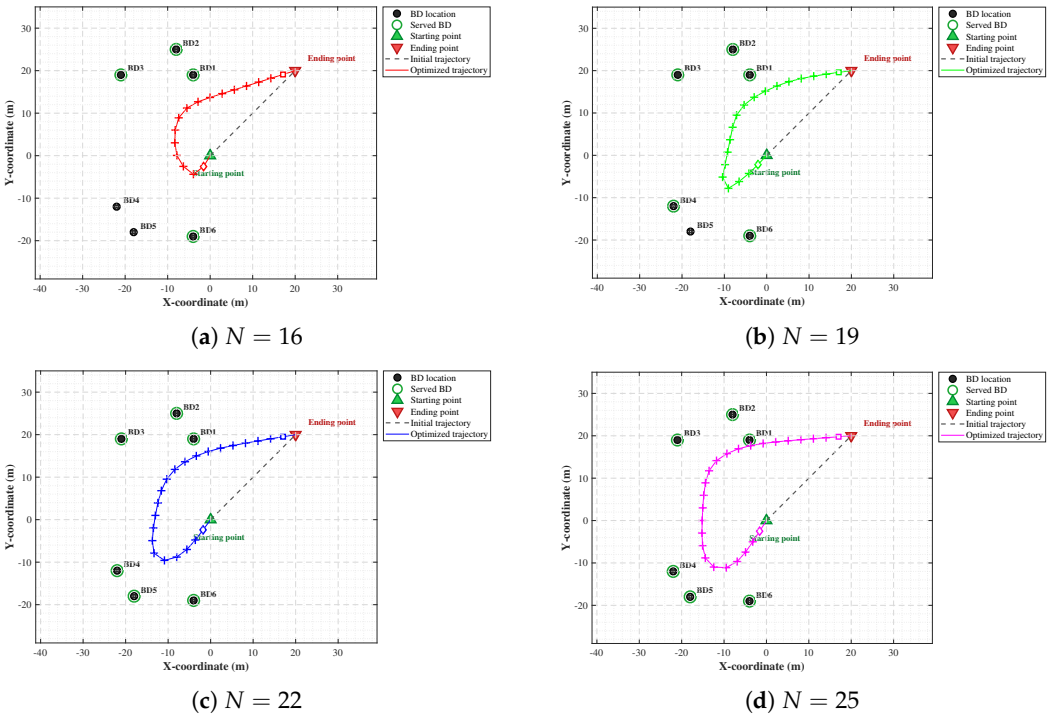


Figure 9. Convergence Behavior of the Proposed Algorithm for $I = 10$ with Homogeneous BD Data Sizes.

Figure 10a–d show the optimal UAV trajectories under different numbers of time slots when the number of BDs is $I = 6$, where the green circles denote the successfully served BDs. It can be observed that, when $N = 16$, the number of time slots is limited and insufficient to support the completion of data transmission for all BDs. Under this condition, while ensuring that it can still reach the destination, the UAV tends to prioritize communication with the more "along-the-way" BDs, namely, BD_1, BD_2, BD_3, and BD_6, which are relatively closer to the destination. When $N = 19$, in addition to successfully serving these more "along-the-way" BDs, the UAV also completes data transmission for BD_4, which is relatively close in distance. When $N = 22$ and $N = 25$, all BDs are successfully served. The difference is that, when $N = 25$, with a more sufficient time budget, the UAV chooses a path closer to BD_1, BD_2, and BD_3 while flying toward the destination, thereby improving the communication quality.

Figure 11a–d illustrate the optimal UAV trajectories under different numbers of time slots when the number of BDs is $I = 10$, where the green circles denote the successfully served BDs. When $N = 24$, BD_3, BD_4, and BD_5 fail to complete data transmission, since they relatively far from both the initial and final locations. When $N = 29$, BD_4 and BD_5 can be successfully served since they are clustered closer to the initial location than BD_3. When $N = 35$ and $N = 40$, all BDs can complete data transmission successfully. In particular, when $N = 40$, the time budget is more sufficient, and the trajectory exhibits a larger curvature, allowing the UAV to approach the left-side BDs, namely, BD_3, BD_4, and BD_5, more closely. As a result, higher communication quality can be achieved compared with the case of $N = 35$.

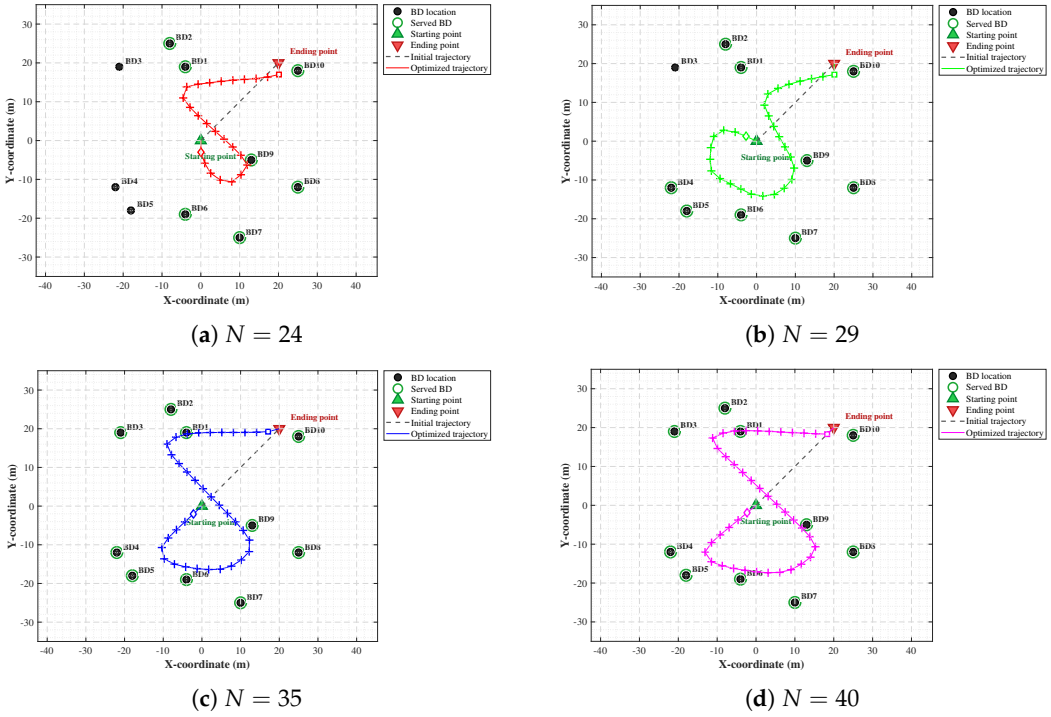
Figures 12 and 13 compare the relaxed solution and the reconstructed solution for the cases of $I = 6$ and $I = 10$, respectively. It can be observed that the reconstructed solution remains close to the relaxed solution. A slightly more noticeable gap appears in the cases of $I = 6$, $N = 19$ and $I = 10$, $N = 29$. This is because, under a limited time budget, the UAV must plan its trajectory to serve as many BDs as possible while ensuring that it can still reach the destination at the final moment. As a result, for some BDs, a large portion of the data has already been transmitted, but the UAV is forced to terminate the transmission before completion in order to satisfy the terminal-position constraint.



(**a**) $N = 16$ (**b**) $N = 19$

(**c**) $N = 22$ (**d**) $N = 25$

Figure 10. Optimized UAV Trajectories with Homogeneous BD Data Sizes and a Straight-Line Initial Path when $I = 6$, (**a**) $N = 16$, (**b**) $N = 19$, (**c**) $N = 22$, (**d**) $N = 25$.



(**a**) $N = 24$ (**b**) $N = 29$

(**c**) $N = 35$ (**d**) $N = 40$

Figure 11. Optimized UAV Trajectories with Homogeneous BD Data Sizes and a Straight-Line Initial Path when $I = 10$, (**a**) $N = 24$, (**b**) $N = 29$, (**c**) $N = 35$, (**d**) $N = 40$.

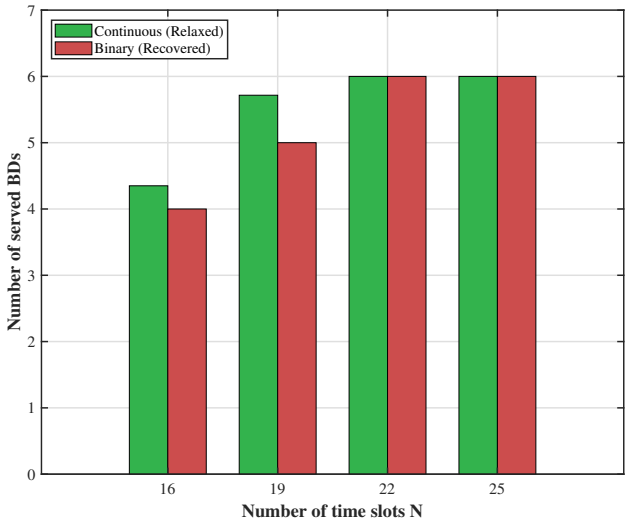


Figure 12. Comparison Between Relaxed and Reconstructed Solutions for $I = 6$ with Homogeneous BD Data Sizes.

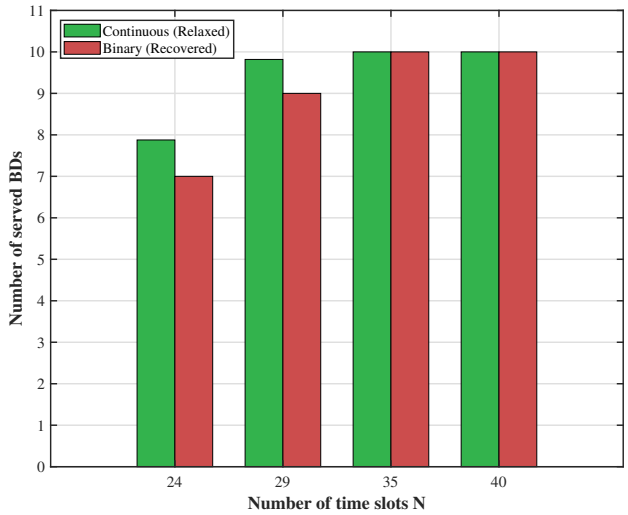


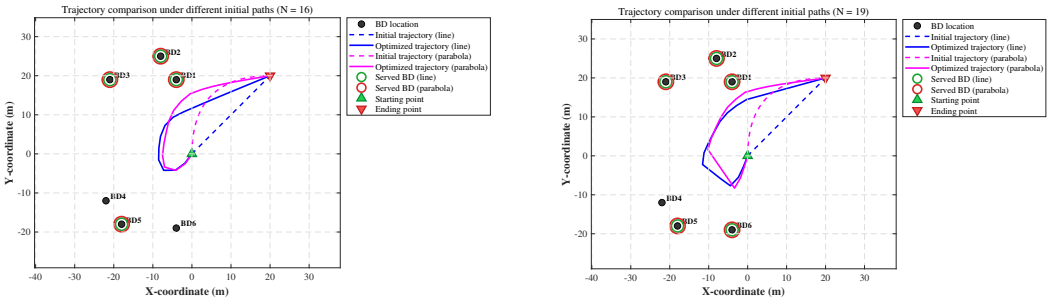
Figure 13. Comparison Between Relaxed and Reconstructed Solutions for $I = 10$ with Homogeneous BD Data Sizes.

4.1.3. Impact of the Initial Path on the Optimization Results

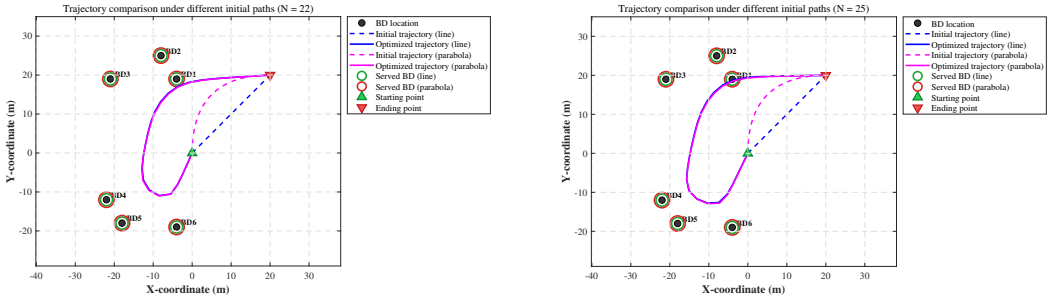
To investigate the impact of the initial path on the optimization results, this subsection replaces the straight-line initial path adopted in Section 4.1.1 with a curved initial path for comparison, while keeping all other parameters the same as those in Section 4.1.1.

As observed from Figure 14a–d, when $I = 6$, although the initial trajectories differ significantly, the iterative optimization process drives them to converge to relatively similar flight paths.

As shown in Figure 15a–d, when $I = 10$, the optimized trajectories under different initial paths exhibit certain differences, indicating that the initial trajectory affects the final optimization result more with larger BD number I. These observation reveals that, although the final trajectories corresponding to different initial paths differ in their specific shapes, they share a consistent overall tendency to perform path adjustments and local detours in the vicinity of several key BDs. This demonstrates that the proposed algorithm is capable of capturing the main direction of optimization and can obtain feasible solutions with favorable performance under different initial conditions.



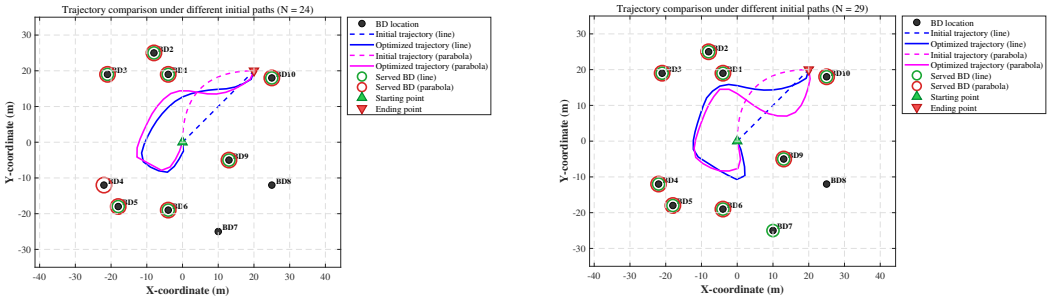
(**a**) $N = 16$ (**b**) $N = 19$



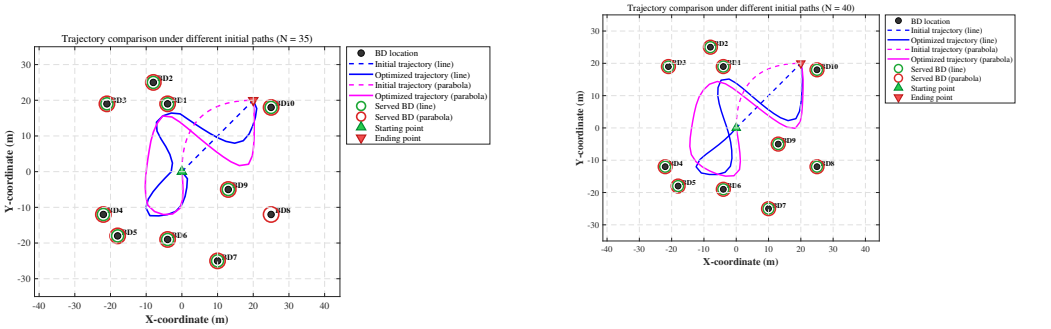
(**c**) $N = 22$ (**d**) $N = 25$

Figure 14. Optimized UAV Trajectories Under Different Initial Paths when $I = 6$, (**a**) $N = 16$, (**b**) $N = 19$, (**c**) $N = 22$, (**d**) $N = 25$.

Therefore, although the initial trajectory may affect the final optimization result to some extent, the proposed BCD-SCA algorithm can still stably converge to high-quality solutions under different initial conditions, which further verifies its effectiveness and robustness in trajectory optimization.



(**a**) $N = 24$ (**b**) $N = 29$



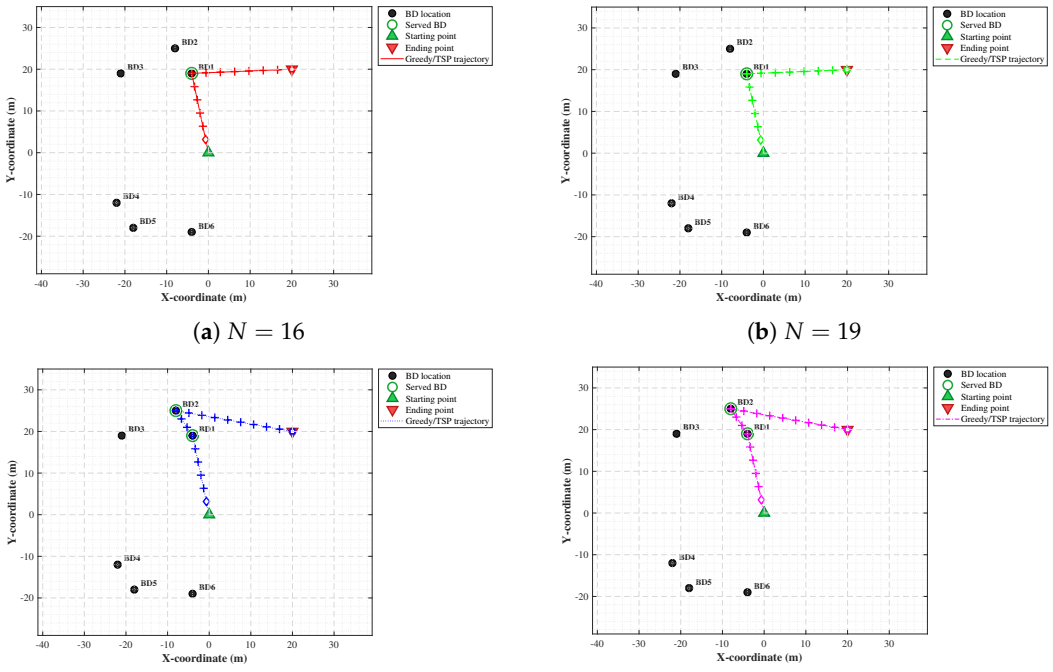
(**c**) $N = 35$ (**d**) $N = 40$

Figure 15. Optimized UAV Trajectories Under Different Initial Paths when $I = 10$, (**a**) $N = 24$, (**b**) $N = 29$, (**c**) $N = 35$, (**d**) $N = 40$.

4.2. Comparison with the TSP-VS and STSP-SO Algorithms

For performance comparison, two benchmark schemes are adopted in this paper, namely, the TSP-based Visit-and-Serve (TSP-VS) scheme and the Smoothed TSP with Scheduling Optimization (STSP-SO) scheme. In the TSP-VS scheme, the visiting order of the BDs is determined by a greedy TSP strategy, and the UAV sequentially flies to each BD location and hovers there to complete data transmission, without performing any scheduling optimization throughout the process. In contrast, in the STSP-SO scheme, after the visiting order is determined, a continuous and smooth UAV flight trajectory is generated via spline interpolation, and communication scheduling is then optimized over this fixed trajectory according to the time-varying channel conditions. The simulations are conducted under the same parameter settings as those used in Section 4.1.1.

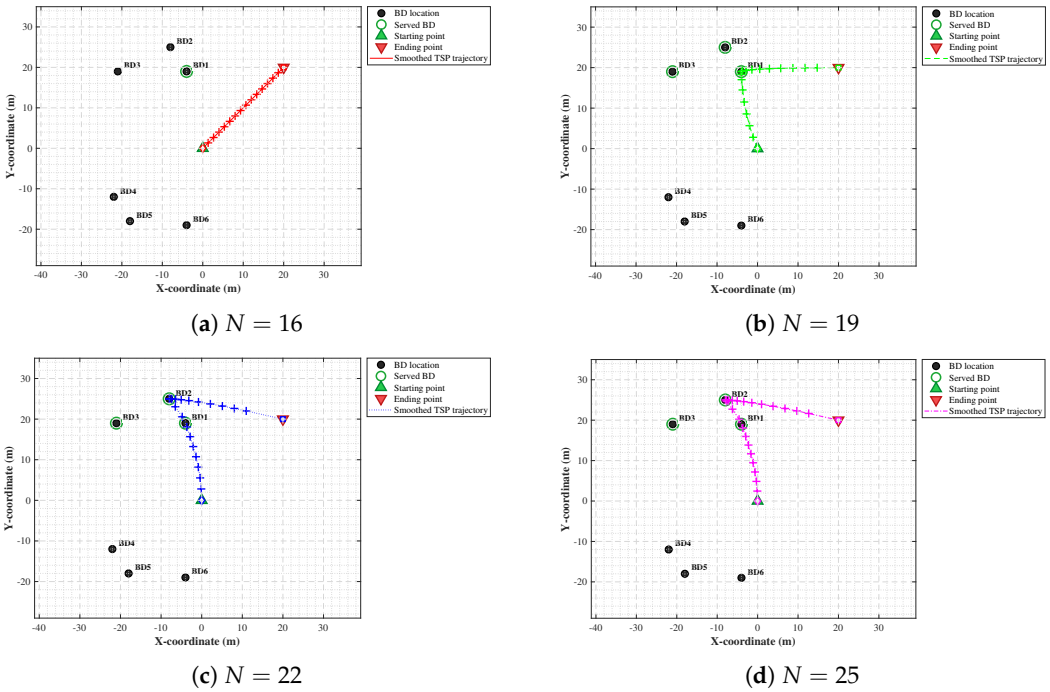
When the number of BDs is $I = 6$, the flight trajectories obtained by the TSP-VS and STSP-SO schemes are shown in Figure 16a–d and Figure 17a–d, respectively. The comparison of the numbers of served BDs achieved by the TSP-VS, STSP-SO, and BCD-SCA algorithms under different numbers of time slots is presented in Figure 18. By comparing these trajectories with those in Figure 4a–d, it can be observed that, under the TSP-VS scheme, the UAV tends to first serve BD_1 at the early stage of the flight because of its relatively short travel distance. However, after completing the service for BD_1, the UAV is still far from BD_4, BD_5, and BD_6. As a result, under a limited time budget, it can subsequently serve only the relatively closer BD_2. For the STSP-SO scheme, when the number of time slots is small, only BD_1 can be successfully served. Even when $N = 25$, it can still serve only BD_1, BD_2, and BD_3, which are closer to the destination. In contrast, under the same time budget, the proposed BCD-SCA algorithm is able to serve all BDs. These results clearly indicate that the advantage of the BCD-SCA algorithm becomes particularly pronounced when the BDs are sparsely distributed.



(**a**) $N = 16$ (**b**) $N = 19$

(**c**) $N = 22$ (**d**) $N = 25$

Figure 16. UAV trajectory under the TSP-VS optimization algorithm when $I = 6$, (**a**) $N = 16$, (**b**) $N = 19$, (**c**) $N = 22$, (**d**) $N = 25$.



(**a**) $N = 16$ (**b**) $N = 19$

(**c**) $N = 22$ (**d**) $N = 25$

Figure 17. UAV trajectory under the STSP-SO optimization algorithm when $I = 6$, (**a**) $N = 16$, (**b**) $N = 19$, (**c**) $N = 22$, (**d**) $N = 25$.

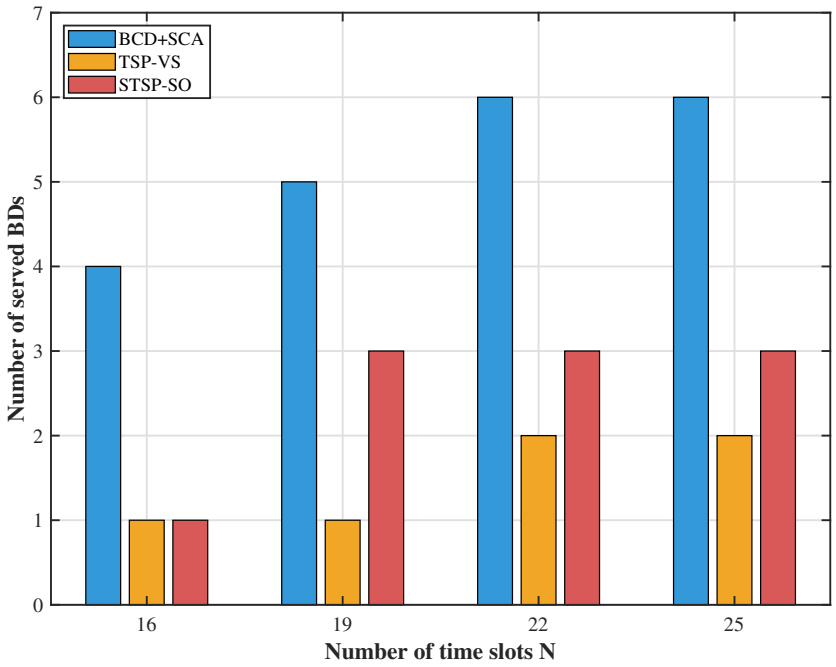
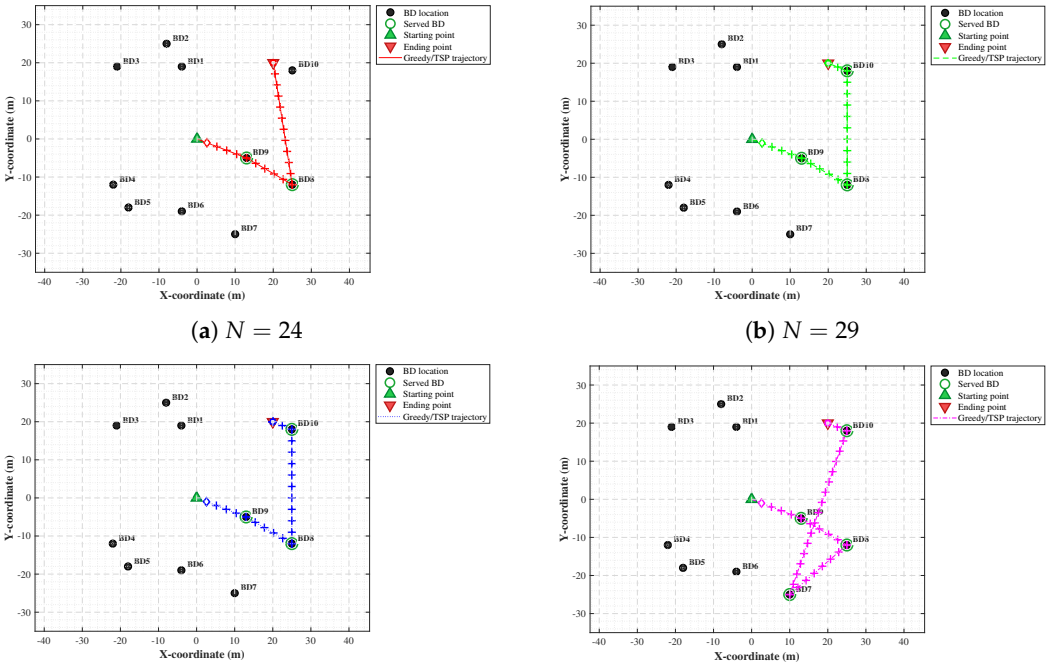


Figure 18. Comparison of the BCD-SCA, TSP-VS, and STSP-SO Algorithms for $I = 6$.

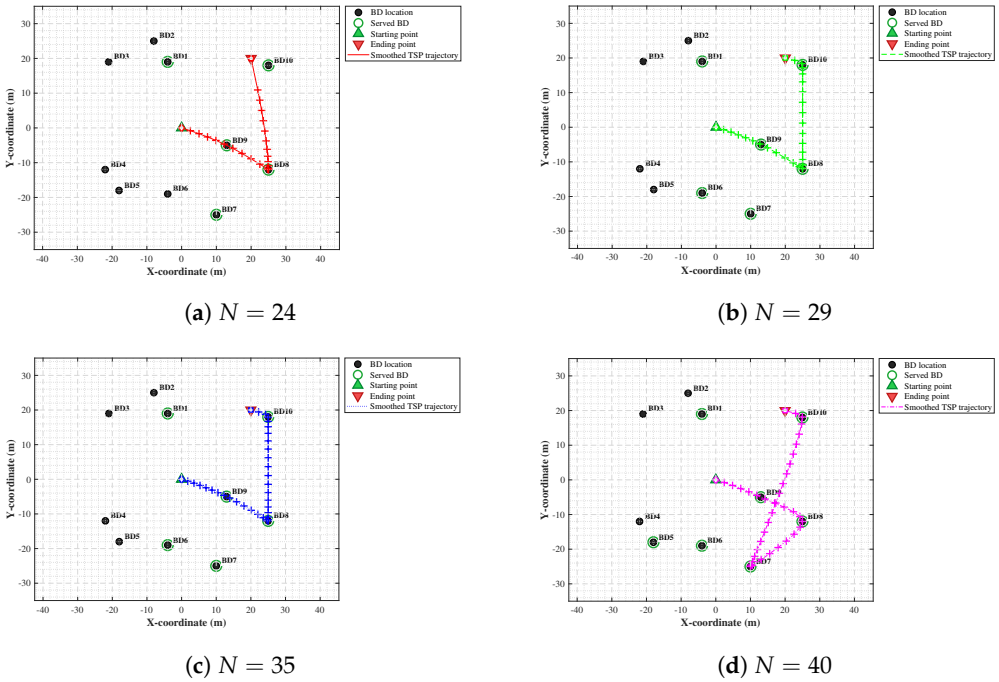
When the number of BDs is $I = 10$, the flight trajectories obtained by the TSP-VS and STSP-SO schemes are shown in Figure 19a–d and Figure 20a–d, respectively. The comparison of the numbers of served BDs achieved by the TSP-VS, STSP-SO, and BCD-SCA algorithms under different numbers of time slots is presented in Figure 21. Compared with the trajectories obtained by the BCD-SCA algorithm in Figure 5a–d, the superiority of the proposed BCD-SCA algorithm is clearly demonstrated.



(**a**) $N = 24$ (**b**) $N = 29$

(**c**) $N = 35$ (**d**) $N = 40$

Figure 19. UAV trajectory under the TSP-VS optimization algorithm when $I = 10$, (**a**) $N = 24$, (**b**) $N = 29$, (**c**) $N = 35$, (**d**) $N = 40$.



(**a**) $N = 24$ (**b**) $N = 29$

(**c**) $N = 35$ (**d**) $N = 40$

Figure 20. UAV trajectory under the STSP-SO optimization algorithm when $I = 10$, (**a**) $N = 24$, (**b**) $N = 29$, (**c**) $N = 35$, (**d**) $N = 40$.

From the perspective of computational complexity, the complexity of the TSP-VS scheme mainly arises from the generation of the traveling path and the service determination under the fixed trajectory. If the TSP-VS algorithm is adopted, its overall computational complexity can generally be approximated as $O(I^2 + IN)$. Building upon the TSP route, the STSP-SO scheme further performs trajectory smoothing and sequential optimization, and

therefore its computational complexity is higher than that of TSP-VS, which can generally be approximated as $O\left(I^2 + (IN)^2\right)$.

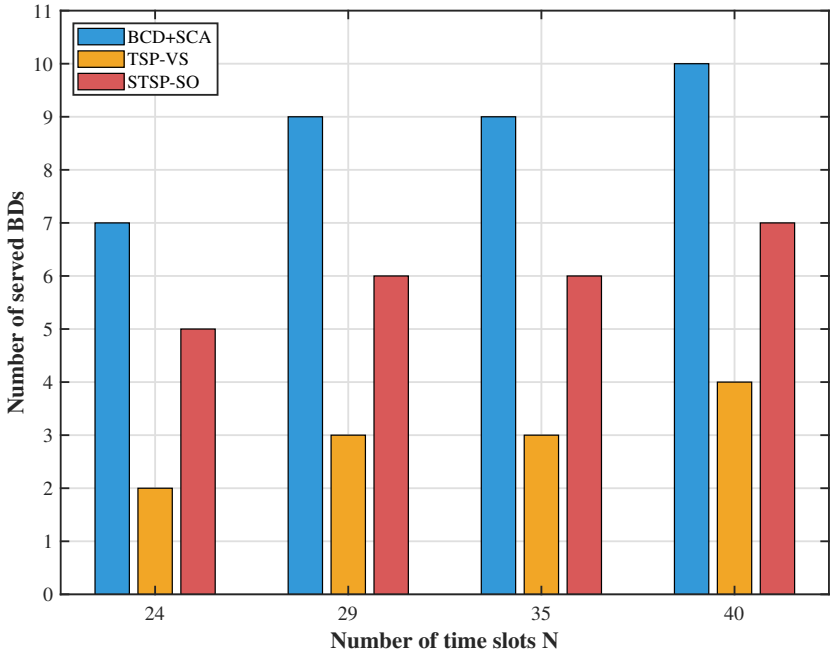


Figure 21. Comparison of the BCD-SCA, TSP-VS, and STSP-SO Algorithms for $I = 10$.

By contrast, the proposed BCD-SCA method alternately solves the resource allocation subproblem and the trajectory optimization subproblem through an outer BCD iteration loop and an inner SCA iteration loop. Its overall computational complexity can be expressed as $O\left(L(J+1)(IN)^3\right)$. From the perspective of computational complexity, the proposed method is therefore more complex than the two benchmark schemes, namely, TSP-VS and STSP-SO. Nevertheless, both TSP-VS and STSP-SO are essentially stage-wise design methods, where the visiting path is determined first, followed by service or resource allocation optimization. As a result, they cannot fully capture the coupling relationship between communication resource allocation and UAV trajectory design. In contrast, the proposed method jointly optimizes communication resources and trajectory design, thereby enabling a more effective improvement in the number of served devices under a given time constraint. Therefore, although the proposed method incurs a higher computational complexity, it achieves substantial performance gains.

5. Conclusions

This paper investigates a UAV-assisted BackCom network for data collection from BDs. Due to the limited mission time, an optimization problem is formulated to jointly optimize communication resource allocation and the UAV trajectory in order to maximize the number of BDs served within the given time. Although the formulated problem is non-convex, it is transformed into a tractable form, and a low-complexity solution is proposed to effectively solve the problem. Simulation results show that, when $I = 6$, the proposed algorithm achieves performance improvements of approximately 250% and 110% over the TSP-VS and STSP-SO schemes, respectively. When $I = 10$, the corresponding performance gains are approximately 190% and 45%, respectively. It should be noted that this paper adopts the free-space path-loss model and assumes a symmetric link structure. Although such assumptions are widely used in UAV-assisted communication studies, in practical scenarios the wireless channel may be affected by factors such as small-scale fading, shadowing, interference, and backscatter efficiency. Therefore, how to perform

joint optimization under more realistic communication environments still deserves further investigation. The proposed method has considerable application potential in time-sensitive IoT scenarios such as environmental monitoring, disaster emergency communications, and smart agriculture. Future work will further consider multi-UAV cooperative data collection, robust optimization under uncertain channel conditions, and joint optimization problems in more complex communication environments.

Author Contributions: Conceptualization, methodology and writing—review and editing, L.S. and W.Z.; formal analysis, and writing—original draft preparation, W.Z.; software, and validation, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: During the preparation of this manuscript, the authors used DeepSeek V3.2 for the purposes of correcting spelling and grammatical errors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Liang, Y.C.; Zhang, Q.; Larsson, E.G.; Li, G.Y. Symbiotic radio: Cognitive backscattering communications for future wireless networks. *IEEE Trans. Cogn. Commun. Netw.* 2020, *early access*. [CrossRef]
2. Lu, W.; Hu, S.; Liu, X.; He, C.; Gong, Y. Incentive mechanism based cooperative spectrum sharing for OFDM cognitive IoT network. *IEEE Trans. Netw. Sci. Eng.* **2020**, *7*, 662–672. [CrossRef]
3. Juwaied, A.; Jackowska-Strumillo, L. Applying Machine Learning to DEEC Protocol: Improved Cluster Formation in Wireless Sensor Networks. *Network* **2025**, *5*, 26. [CrossRef]
4. Aslam, N.; Xia, K.; Hadi, M.U. Optimal Wireless Charging Inclusive of Intellectual Routing Based on SARSA Learning in Renewable Wireless Sensor Networks. *IEEE Sens. J.* **2019**, *19*, 8340–8351. [CrossRef]
5. Bagwari, A.; Logeshwaran, J.; Usha, K.; Raju, K.; Alsharif, M.H.; Uthansakul, P.; Uthansakul, M. An Enhanced Energy Optimization Model for Industrial Wireless Sensor Networks Using Machine Learning. *IEEE Access* **2023**, *11*, 96343–96362. [CrossRef]
6. Juwaied, A. EMO-PEGASIS: A Dual-Phase Machine Learning Protocol for Energy Delay Optimisation in WSNs. *Sensors* **2026**, *26*, 611. [CrossRef]
7. Bletsas, A.; Alevizos, P.N.; Vougioukas, G. The Art of Signal Processing in Backscatter Radio for μW (or Less) Internet of Things: Intelligent Signal Processing and Backscatter Radio Enabling Batteryless Connectivity. *IEEE Signal Process. Mag.* **2018**, *35*, 28–40. [CrossRef]
8. Boyer, C.; Roy, S. Backscatter communication and RFID: Coding, energy, and MIMO analysis. *IEEE Trans. Commun.* **2014**, *62*, 770–785. [CrossRef]
9. Xu, C.; Yang, L.; Zhang, P. Practical backscatter communication systems for battery-free Internet of Things: A tutorial and survey of recent research. *IEEE Signal Process. Mag.* **2018**, *35*, 16–27. [CrossRef]
10. Liu, V.; Parks, A.; Talla, V.; Gollakota, S.; Wetherall, D.; Smith, J.R. Ambient backscatter: Wireless communication out of thin air. In Proceedings of the ACM SIGCOMM 2013 Conference, Hong Kong, China, 12–16 August 2013; pp. 1–13. [CrossRef]
11. Yang, G.; Liang, Y.C.; Zhang, R.; Pei, Y. Modulation in the air: Backscatter communication over ambient OFDM carrier. *IEEE Trans. Commun.* **2018**, *66*, 1219–1233. [CrossRef]
12. Kimionis, J.; Bletsas, A.; Sahalos, J.N. Increased range bistatic scatter radio. *IEEE Trans. Commun.* **2014**, *62*, 1091–1104. [CrossRef]
13. Fasarakis-Hilliard, N.; Alevizos, P.N.; Bletsas, A. Coherent detection and channel coding for bistatic scatter radio sensor networking. *IEEE Trans. Commun.* **2015**, *63*, 1798–1810. [CrossRef]
14. Varshney, A.; Harms, O.; Pérez-Penichet, C.; Rohner, C.; Hermans, F.; Voigt, T. LoRea: A Backscatter Architecture That Achieves a Long Communication Range. In Proceedings of the 15th ACM Conference Embedded Network Sensor Systems, (SenSys), Delft, The Netherlands, 6–8 November 2017; pp. 1–14. [CrossRef]

15. Gong, S.; Huang, X.; Xu, J.; Liu, W.; Wang, P.; Niyato, D. Backscatter relay communications powered by wireless energy beamforming. *IEEE Trans. Commun.* **2018**, *66*, 3187–3200. [CrossRef]
16. Yang, G.; Xu, X.; Liang, Y.C. Resource Allocation in NOMA-Enhanced Backscatter Communication Networks for Wireless Powered IoT. *IEEE Wireless Commun. Lett.* **2020**, *9*, 117–120. [CrossRef]
17. Zeng, Y.; Xu, J.; Zhang, R. Energy Minimization for Wireless Communication with Rotary-Wing UAV. *IEEE Trans. Wireless Commun.* **2019**, *18*, 2329–2345. [CrossRef]
18. Han, R.; Bai, L.; Wen, Y.; Liu, J.; Choi, J.; Zhang, W. UAV-Aided Backscatter Communications: Performance Analysis and Trajectory Optimization. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3129–3143. [CrossRef]
19. Farajzadeh, A.; Ercetin, O.; Yanikomeroglu, H. UAV Data Collection over NOMA Backscatter Networks: UAV Altitude and Trajectory Optimization. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019. [CrossRef]
20. Mozaffari, M.; Saad, W.; Bennis, M.; Nam, Y.H.; Debbah, M. A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2334–2360. [CrossRef]
21. Zhang, S.; Zhang, H.; Di, B.; Song, L. Cellular UAV-to-X Communications: Design and Optimization for Multi-UAV Networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 1346–1359. [CrossRef]
22. Lyu, J.; Zeng, Y.; Zhang, R.; Lim, T.J. Placement Optimization of UAV-Mounted Mobile Base Stations. *IEEE Commun. Lett.* **2017**, *21*, 604–607. [CrossRef]
23. Hua, M.; Yang, L.; Li, C.; Wu, Q.; Swindlehurst, A.L. Throughput Maximization for UAV-Aided Backscatter Communication Networks. *IEEE Trans. Commun.* **2020**, *68*, 1254–1270. [CrossRef]
24. Yang, G.; Dai, R.; Liang, Y.C. Energy-Efficient UAV Backscatter Communication with Joint Trajectory Design and Resource Optimization. *IEEE Trans. Wireless Commun.* **2021**, *20*, 926–941. [CrossRef]
25. Xie, P.; Jiang, J.; Li, T.; Lu, Y. Joint Optimization of Resource Allocation and Trajectory Based on User Trajectory for UAV-Assisted Backscatter Communication System. *China Commun.* **2024**, *21*, 197–209. [CrossRef]
26. Samir, M.; Sharafeddine, S.; Assi, C.M.; Nguyen, T.M.; Ghrayeb, A. UAV Trajectory Planning for Data Collection from Time-Constrained IoT Devices. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 34–46. [CrossRef]
27. Yuan, X.; Hu, Y.; Zhang, J.; Schmeink, A. Joint User Scheduling and UAV Trajectory Design on Completion Time Minimization for UAV-Aided Data Collection. *IEEE Trans. Wirel. Commun.* **2023**, *22*, 3884–3898. [CrossRef]
28. Al-Hilo, A.; Samir, M.; Elhattab, M.; Assi, C.; Sharafeddine, S. RIS-Assisted UAV for Timely Data Collection in IoT Networks. *IEEE Syst. J.* **2023**, *17*, 431–442. [CrossRef]
29. Tseng, P. Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization. *J. Optim. Theory Appl.* **2001**, *109*, 475–494. [CrossRef]
30. Beck, A.; Ben-Tal, A.; Tetruashvili, L. A Sequential Parametric Convex Approximation Method with Applications to Non-Convex Truss Topology Design Problems. *J. Glob. Optim.* **2010**, *47*, 29–51. [CrossRef]
31. Wu, Q.; Zeng, Y.; Zhang, R. Joint Trajectory and Communication Design for Multi-UAV Enabled Wireless Networks. *IEEE Trans. Wireless Commun.* **2018**, *17*, 2109–2121. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.