Article

Shooting Utility Maximization in UAV-Assisted Wireless Camera Sensor Networks

Yulei Wu 1, Simeng Feng 1,*, Chao Dong 1,*, and Weijun Wang 2

1 College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China; wuyulei@nuaa.edu.cn
2 College of Computer Science and Technology, Nanjing University, Nanjing 210023, China; weijunwang@mail.nju.edu.cn
* Correspondence: simeng-feng@nuaa.edu.cn (S.F.); dch@nuaa.edu.cn (C.D.)

Abstract: Recently, wireless camera sensor networks (WCSNs) have entered an era of rapid development, and WCSNs assisted by unmanned aerial vehicles (UAVs) are capable of providing enhanced flexibility, robustness and efficiency when executing missions such as shooting targets. Existing research has mainly focused on back-end image processing to improve the quality of captured images, but it has neglected the question of attaining quality images on the front-end, which is significantly influenced by the location and hovering time of the UAV. Therefore, in this paper, we conceive a novel shooting utility model to quantify shooting quality, which is maximized by simultaneously considering the UAV’s trajectory planning, hovering time and shooting point selection. To expound further, we prove the submodularity of the utility function, whereby the original problem can be expressed as a submodular maximization problem with path constraints, and we propose a utility-cost ratio (UCR) algorithm to maximize shooting utility through two-level optimization. Then, by using the relaxation of the cost function, we analyze the gap between the proposed algorithm and the optimal algorithm (OPT) and prove that the UCR algorithm has a bi-criterion approximation ratio of $(1 - 1/e)/2$. Simulation results show that the algorithm outperforms both the random algorithm (RAN) and the maximum shooting utility point selection algorithm (MSU) in terms of shooting utility and time utilization efficiency, improving shooting utility by 51% and 21% compared to the RAN and MSU algorithms, respectively, and achieving at least 88.2% of the OPT algorithm in terms of time utilization efficiency.

Keywords: unmanned aerial vehicle (UAV); target shooting utility; trajectory planning; energy consumption

1. Introduction

In recent years, wireless camera sensor networks (WCSNs) have attracted substantial attention from academia and industry, as they can obtain high-resolution target information by providing videos and images [1–3]. WCSNs are capable of supporting a wide range of applications, such as traffic monitoring [4,5], building inspection [6] and crowd protection [7], with the aid of camera shooting. However, setting up fixed camera sensors for WCSNs may require extensive infrastructure implementation, resulting in cost inefficiencies.

Fortunately, thanks to the mobility, flexibility and economic feasibility of unmanned aerial vehicles (UAVs), these devices constitute promising candidates for shooting images for WCSNs when the UAVs are equipped with camera sensors [8–11]. As shown in Figure 1, when shooting and monitoring urban buildings, a traditional fixed camera sensor network cannot cope with challenges such as high altitude or panoramic shooting, whereas UAVs can move toward targets for multi-angle shooting, which greatly improves the efficiency of shooting tasks.
Substantial efforts have been made in the research of UAV-assisted WCSNs, and the existing research mainly focuses on conserving system resources and improving image quality. First, in terms of conserving system resources, some works are mainly devoted to the question of how to reduce the energy and power consumption of UAVs. For example, Gao et al. [12] designed a network of UAV-assisted camera sensors to detect air quality. In order to minimize the energy consumption of the UAVs, they optimized the flight altitude of the UAVs and their two-dimensional coordinates when flying with the aim of ensuring the accuracy of air quality indicators. Ji et al. [13] investigated the problem of using cooperative UAVs in reconnaissance missions to achieve full coverage of targets. They achieved fairness in energy consumption by planning the trajectory of multiple UAVs. Zhan et al. [14] considered the energy consumption problem when using UAVs to collect data from sensor nodes. They jointly optimized the wake-up time of sensor nodes and the trajectory of the UAVs to minimize the maximum energy consumption of all sensor nodes.

![Figure 1. A diagram of a UAV-assisted WCSN scenario.](image)

In addition, some works mainly focus on improving image quality. Considering the problem of low illumination in UAV pedestrian detection at night, Wang et al. [15] proposed a block-matching and 3D-filtering method to improve the low-illumination image quality. Yuan et al. [16] proposed a seam-cutting strategy based on UAV image mosaics to reduce image distortions and artifacts. Zhang et al. [17] proposed a new real-time pure image mosaic technology for UAVs. They synthesized aerial images through mosaic transformation to obtain wide-angle panoramic images quickly and accurately. Wang et al. [18] considered the scenario of blurred images due to insufficient power when UAVs shoot high-frequency images and proposed a nonlinear image enhancement method to improve the quality of blurred images from UAV aerial photography. However, these studies all focused on back-end image processing, and the influence of the flight trajectory and the hovering time of the UAV on shooting was not considered at the front-end.

As far as we know, most of the existing research on UAV-assisted WCSNs focuses on the back-end processing of UAV-shot images. However, there is little consideration given to the front-end, such as the influence of the shooting time and the hovering position of the UAV on the shot image. With respect to UAV-assisted data acquisition, we know that the hovering position and hovering time of UAVs have a significant influence on the information collected [19,20]. Based on this principle, in this paper, we conceive the concept of a shooting utility to quantify the amount of information obtained. Then, the UAV’s trajectory planning, hovering time and shooting point selection are optimized to maximize shooting utility. The main contributions of this work are as follows:
We conceive a novel shooting utility model to quantify shooting quality and then address the problem of maximizing shooting utility. We show that the elaborated shooting utility function obeys the rule of submodularity, which is beneficial for transforming the original problem into a submodularity maximization problem with path constraints.

We formulate a utility-cost ratio (UCR) algorithm to maximize shooting utility. By using the relaxation of the cost function, we analyze the gap between the proposed algorithm and the optimal algorithm (OPT) and prove that the UCR algorithm has a bi-criterion approximation ratio of \((1 - 1/e)/2\).

We conduct numerous simulations to analyze the performance of the UCR algorithm. The results demonstrate that the algorithm outperforms both the random algorithm (RAN) and the maximum shooting utility point selection algorithm (MSU) in terms of shooting utility and time utilization efficiency, improving shooting utility by 51% and 21% compared to the RAN and MSU algorithms, respectively, and achieving at least 88.2% of the OPT algorithm in terms of time utilization efficiency.

The rest of this paper is organized as follows. In Section 2, we describe the model of the UAV-enabled shooting system in detail. In Section 3, we analyze the problem and put forward a solution. Section 4 provides the simulation results and performance evaluation. Finally, we summarize this paper in Section 5.

### 2. System Model

We consider a UAV-enabled shooting mission scenario, as shown in Figure 2. Assume that we have a total of \(M\) targets \(O = \{o_1, o_2, \ldots, o_M\}\) on the ground that must be shot. A UAV equipped with a camera is capable of shooting an area with radius \(R\). Since the ground targets are widely distributed, it is expensive, time-consuming and difficult for the UAV to fly over each target for its shooting mission. Therefore, we adopt an overlapping clustering approach [21,22] to group the ground targets into different clusters to accomplish the shooting mission efficiently, and the cluster head set is denoted as \(C = \{c_1, c_2, \ldots, c_n\}\). By doing so, the UAV is capable of moving to the position above the cluster head to complete the shooting of members in the cluster, which means that the UAV can capture multiple targets at a single shooting point, and a target can be captured at multiple shooting points. We study the scenario of a UAV flying at altitude \(h\) with a fixed speed \(v_a\). The UAV starts at the service station \(s_0\) and returns to \(s_0\) after completing the shooting tasks. The major notations used in this paper are summarized in Table 1.
Table 1. Notations.

| Notation | Definition                      |
|----------|---------------------------------|
| O        | Ground target set               |
| C        | Cluster head set                |
| L        | UAV trajectory set              |
| m        | Number of ground targets        |
| n        | Number of cluster heads         |
| $d(c_i, c_{i+1})$ | Distance between cluster head $i$ and $i + 1$ |
| $E_m$    | Motion energy cost of UAV       |
| $E_h$    | Hovering energy cost of UAV     |
| $E_{max}$| Energy capacity of UAV          |
| $v_a$    | Velocity of UAV                 |
| R        | Shooting radius of UAV          |
| T        | Time allocation strategy        |
| K        | Selected cluster head set       |
| $Q(K, o_j)$ | Shooting utility for single target $o_j$ |
| $Q(K, T)$ | Overall Shooting utility        |

2.1. Energy Consumption Model

We divide the energy consumption model into two categories: one is motion energy cost, and the other is hovering energy cost. The motion energy is related to the flight distance of the UAV between clusters, while the hovering energy is related to the hovering time of the UAV at each cluster head. We can use $(x(c_i), y(c_i))$ to denote the location coordinates of the cluster head $c_i$; then, the Euclidean distance between two cluster heads can be expressed as

$$d(c_i, c_{i+1}) = \sqrt{[x(c_{i+1}) - x(c_i)]^2 + [y(c_{i+1}) - y(c_i)]^2}. \quad (1)$$

We use the set $K$ to represent the cluster heads over which the UAV chooses to hover; then, a new selected cluster head set $K \subseteq C$ can be formed and used to obtain the following motion energy cost

$$E_m = \sum_{d(k_i, k_{i+1}) \in L_{TSP}(K)} \frac{p_m d(k_i, k_{i+1})}{\eta v_a}, \quad (2)$$

where $d(k_i, k_{i+1})$ represents the distance between two adjacent points in the set $K$, $L_{TSP}(K)$ is a closed path for the UAV that starts at and returns to the service station $s_0$, $p_m$ is the minimum motion power consumption of the UAV and $\eta$ is the UAV’s power efficiency. We use the parameter $\lambda$ to represent $p_m / \eta v_a$. The UAV only visits each cluster head in $K$ once.

The UAV’s hovering energy consumption is related to the hovering time and various parameters of the UAV. We use $t_i$ to denote the time that UAV hovers at the cluster head $c_i$. Then, we denote the hovering energy consumption for cluster head $c_i$ as

$$E(c_i) = \sqrt{\frac{(mg)^3}{2\pi\rho r_n^3 n_u} t_i}, \quad (3)$$

where $m$, $g$ and $\rho$ are, respectively, the mass of the UAV, the gravity of the earth and the air density. The radius of the rotor and the number of rotors are expressed by $r_n$ and $n_u$, respectively.
respectively. We use parameter $\mu$ to represent $\sqrt{\frac{(mg)^3}{2\pi \rho r^2 \eta}}$. For the selected cluster head set $\mathcal{K}$, the total hovering energy consumption can be expressed as

$$E_h = \sum_{c_i \in \mathcal{K}} E(c_i) = \sum_{c_i \in \mathcal{C}} \sqrt{\frac{(mg)^3}{2\pi \rho r^2 \eta}} t_i a_i,$$

where $a_i$ is a binary variable used to indicate whether the UAV chooses to hover above the cluster head $c_i$ for shooting. For example, $a_i = 1$ if $c_i$ is selected; otherwise, it is 0. Thus, the total cost for any selected cluster head set $\mathcal{K}$ can is given by

$$E(\mathcal{K}) = E_m + E_h$$

$$= \sum_{d \in \mathcal{E}_{TSP}(\mathcal{K})} \frac{p_{md}}{\eta \eta_u} + \sum_{c_i \in \mathcal{C}} \sqrt{\frac{(mg)^3}{2\pi \rho r^2 \eta_u}} t_i a_i,$$

where $a_i$ is a binary variable used to indicate whether the UAV chooses to hover above the cluster head $c_i$ for shooting. For example, $a_i = 1$ if $c_i$ is selected; otherwise, it is 0. Thus, the total cost for any selected cluster head set $\mathcal{K}$ can is given by

$$E(\mathcal{K}) = E_m + E_h$$

$$= \sum_{d \in \mathcal{E}_{TSP}(\mathcal{K})} \frac{p_{md}}{\eta \eta_u} + \sum_{c_i \in \mathcal{C}} \sqrt{\frac{(mg)^3}{2\pi \rho r^2 \eta_u}} t_i a_i.$$

### 2.2. Shooting Utility Model

As the shooting time is related to acquired information, we formulate the shooting utility model to quantify the amount of information we obtain. For the shooting utility model, we first define the shooting utility weight $p_{ij}$, that is, the amount of information that can be obtained per unit time by shooting a cluster member $o_j$ above the cluster head $c_i$. Then we find that the shooting utility of the UAV flying above the cluster head $c_i$ on target $o_j$ is

$$q(o_j, c_i) = p_{ij} t_{ij},$$

where $t_{ij}$ is the shooting time allocated to the target $o_j$ when the UAV hovers over the cluster head $c_i$. According to overlapping clustering, a target may belong to multiple clusters, and the utility of a target contributed by multiple clusters is additive. That is to say, for any given set $\mathcal{K} \subseteq \mathcal{C}$, the total shooting utility for target $o_j$ can be expressed as

$$q_{\mathcal{K}}(o_j) = \sum_{c_i \in \mathcal{K}} q(o_j, c_i) = \sum_{c_i \in \mathcal{C}} p_{ij} t_{ij} a_i.$$

Furthermore, considering the varying importance of each target, the maximum amount of information that can be captured also varies, so that there is a maximum shooting utility $Q^m_{ij}$ defined for each target. When the shooting time is greater than a predefined threshold, shooting utility remains equal to $Q^m_{ij}$. Thus, we can obtain the complete shooting utility of the target $o_j$ as

$$Q(\mathcal{K}, o_j) = \min \left\{ q_{\mathcal{K}}(o_j), Q^m_{ij} \right\}.$$

In order to simplify the model, in this paper, we do not consider the correlation between targets or the information redundancy when shooting the same target from different positions. We think that shooting utility is a simple addition quantity, but in this work, we consider it according to the feature distribution of computer vision and use information theory modeling to solve this problem [23].

### 3. Problem Formulation

The total shooting utility is the sum of the shooting utility of UAVs for all members in the selected clusters, which can be expressed as

$$Q(\mathcal{K}, T) = \sum_{j=1}^{m} Q(\mathcal{K}, o_j) = \sum_{j=1}^{m} \min \left\{ q_{\mathcal{K}}(o_j), Q^m_{ij} \right\} = \sum_{j=1}^{m} \min \left\{ \sum_{c_i \in \mathcal{C}} p_{ij} t_{ij} a_i, Q^m_{ij} \right\}. $$


in which $\mathcal{T} = \{t_{ij} | i = 1, \ldots, n; j = 1, \ldots, m\}$ represents the shooting time allocation of the UAV to targets in different clusters. As the sum of shooting time assigned to cluster members cannot be greater than the hover time of the UAV at the cluster head position, we have

$$\sum_{o_j \in O} t_{ij} a_i \leq t_i, \forall c_i \in \mathcal{C}. \quad (10)$$

At the same time, considering the energy constraints of the UAV, the total energy consumption cannot exceed its maximum energy consumption, which can be described as

$$E(\mathcal{K}) \leq E_{\text{max}}, \forall \mathcal{K} \subseteq \mathcal{C}, \quad (11)$$

where $E_{\text{max}}$ is the maximum energy consumption of the UAV.

As defined above, our goal is to find a closed path for the UAV, that is, to find a cluster head subset $\mathcal{K}$ that maximizes the shooting utility of the UAV on a target in the network under the conditions expressed in Equations (10) and (11). Thus we can describe our problem as

$$P1 : \max_{a_i, t_{ij}} Q(\mathcal{K}, \mathcal{T})$$

s.t. $\sum_{o_j \in O} t_{ij} a_i \leq t_i, \forall c_i \in \mathcal{C}, \quad (12)$

$$E(\mathcal{K}) \leq E_{\text{max}}, \forall \mathcal{K} \subseteq \mathcal{C}.$$  

The cluster head set $\mathcal{K}$ and time allocation strategy $\mathcal{T}$ can be found by optimizing $a_i$ and $t_{ij}$.

According to the above definition, the formulated problem consists of a two-layer optimization, where the optimization objective at the bottom layer is cluster head selection and time allocation, and the top layer performs trajectory planning for the energy-constrained UAV based on the selected cluster heads. The two layers of optimization are interrelated and cannot be handled separately to obtain the global solution. If the cluster head is selected without considering trajectory planning, a lot of motion energy is consumed. If the appropriate cluster head is not considered in trajectory planning, the shooting utility cannot be maximized.

When considering only shooting utility without considering the UAV energy constraint, we can reduce the problem $P1$ to a 0-1 backpack problem, as follows:

$$\max_{a_i, t_{ij}} Q(\mathcal{K}, \mathcal{T})$$

s.t. $\sum_{o_j \in O} t_{ij} a_i \leq t_i, \forall c_i \in \mathcal{C}, \quad (13)$

The 0-1 backpack problem is a classic NP-hard problem. When we further consider the energy consumption of the UAV, the problem can be modeled as a traveling salesman problem (TSP). It is intractable to attain the traveling cost $E(\mathcal{K})$, as it needs to find a closed TSP tour $\mathcal{L}_{\text{TSP}}(\mathcal{K})$, which is also an NP-hard problem. Therefore, the formulated optimization problem $P1$ is coupled by two NP-hard problems.

4. An Effective Approximation Algorithm: UCR Algorithm

In this section, we first prove the submodularity of the utility function, based on which we can formulate the problem as a submodular maximization problem with path constraints and propose an efficient approximation algorithm, that is, a UCR, to maximize the shooting utility of the UAV.
4.1. A Proposed UCR Algorithm

In order to solve our problem, we first prove that our function has three properties: monotonicity, nonnegativity and submodularity. These properties are beneficial because they allow us to transform our problem into a submodule maximization problem with path constraints. For convenience of expression, we construct a function $G(K)$ to represent our shooting utility function. We obtain the equational relation $G(K) = Q(K, T')$, which means that when the time allocation strategy is $T'$, the total shooting utility is $G(K)$. Then we prove that the function $G(K)$ has the above three properties.

**Theorem 1.** The constructed function $G(K)$ is monotone, nonnegative and submodular.

**Proof.** The detailed procedure is shown in Appendix A. □

On the basis of proving that $G(K)$ is a submodule function, we design an effective algorithm that considers two-level optimization to solve the submodule maximization problem with path constraints by referring to the idea in [24,25].

For the cluster head $c_i$ in $C$, we sort the ground targets in order of decreasing utility weights. The UAV then greedily allocates the shooting time to each target in this order until each target rises to the utility threshold or the UAV has no remaining time to allocate at the cluster head $c_i$ position. That is to say, before shooting utility reaches the threshold, we tend to allocate more time to shoot targets with higher utility weights. Furthermore, we need to take into account the energy constraints of the UAV. In each iteration, we use the nearest neighbor rule to calculate the energy consumption of the UAV. The core of this algorithm is that for a given set $K$, we need to add a new cluster head $c_i$ with the maximum utility-cost ratio to the current set. The newly added cluster head can be represented as

$$c_i = \arg\max_{c \in C \setminus K_{i-1}} \frac{G(K_{i-1} + c) - G(K_{i-1})}{E(K_{i-1} + c) - E(K_{i-1})},$$  \hspace{1cm} (14)

where $\hat{E}(\cdot)$ is an approximate cost function of $E(\cdot)$. The reason for this substitution is that we observe that the optimal cost function cannot be calculated in polynomial time, so we use an effective approximate cost function in its place. This may influence the quality of our results. Therefore, in the next section, we prove the approximation ratio of the algorithm.

The detailed steps of the UCR algorithm can be found in Algorithm 1. In each iteration, we use the greedy method to select a cluster head with the largest utility-cost ratio. Using this iterative method, we can continuously add new cluster heads to the set $K$ to obtain higher shooting utility until the energy consumption constraint of the UAV is broken.

### 4.2. Approximation Analysis

One of the major limitations of prior works on submodular maximization with routing constraints is that the cost function was usually assumed to be a submodule. However, in many practical problems, the optimal cost function is not submodular. To solve this problem, we take advantage of the natural relaxation of the cost function, the $\beta$-submodularity [26], and the approximation of our algorithm is affected by the $\beta$-submodularity.
**Definition 2.** For the energy consumption cost function $E$, \( \beta \)-submodularity is defined as

\[
\beta = \min_c \min_{A:B \subseteq C} \frac{E(A + c) - E(A)}{E(B + c) - E(B)}.
\]  

(15)

**Algorithm 1 Utility-Cost Ratio Algorithm**

**Input:** Cluster head set $C = \{c_1, c_2, \ldots, c_n\}$, target set $O = \{o_1, o_2, \ldots, o_m\}$, energy constraint $\hat{E}_{\text{max}}$, shooting utility weight $p_j$.

**Output:** Selected cluster heads $K \subseteq C$, time allocation $T$.

1. Initialize $C' \leftarrow C$, $T \leftarrow \emptyset$, $K_0 \leftarrow \emptyset$, $\hat{E}(K_0) \leftarrow 0$, $i \leftarrow 1$, $K' = \arg\max\{G(c) | c \in C, \hat{E}(c) \leq E_{\text{max}}\}$;
2. while $C' \neq \emptyset$
3. for $c \in C'$ do
4. Calculate $G(K_{i-1} + c)$ and $G(K_{i-1})$ with corresponding time allocation scheme;
5. Use the nearest neighbour rule to calculate the energy consumption $\hat{E}(K_{i-1} + c)$ and $\hat{E}(K_{i-1})$;
6. end for
7. \( c' = \arg\max_{c \in C_{K_{i-1}}} G(K_{i-1} + c) - G(K_{i-1}) \);
8. if $\hat{E}(K_{i-1} + c') \leq E_{\text{max}}$ then
9. \( K_i \leftarrow K_{i-1} + c', i \leftarrow i + 1, T \leftarrow T_c' \);
10. end if
11. \( C' \leftarrow C' \setminus c' \);
12. end while
13. if $G(K') \geq G(K_{i-1})$ then
14. $K \leftarrow K'$;
15. else
16. $K \leftarrow K_{i-1}$;
17. end if
18. Output $K \subseteq C, T$

In addition, for the cost function $E$, we need another concept curvature to help prove the approximate ratio, which essentially measures the linear deviation.

**Definition 3.** For the cost function $E$ in the subsets of $\Psi$, we can define the total curvature $\overline{K}_c$ as

\[
\overline{K}_c = 1 - \min_{c \in \Psi} \frac{E(\Psi \setminus c + c) - E(\Psi)}{E(c)}.
\]  

(16)

**Definition 4.** We introduce another variable $N_c$, which is the largest set of feasible solutions for our problem in the set $\Psi$, defined as

\[
N_c = \max\{|\mathcal{K}| : E(\mathcal{K}) \leq E_{\text{max}}\}.
\]  

(17)

In the UCR algorithm, the initial set is $K_0 = 0$; through iteration, we can continuously add new cluster heads to the set $K$, thus generating a series of intermediate sets $K_1, \ldots, K_s$ until the energy consumption constraint of UAV is broken. We assume that the $s + 1$ iteration does not satisfy the constraint, and then the iteration stops.

**Lemma 1.** For $i = 1, \ldots, s + 1$, it holds that

\[
G(K_i) \geq \left[ 1 - \prod_{l=1}^{i} \left(1 - \frac{\hat{E}(K_i)}{\hat{E}(K_{i-1})} \right) \right] G(K_i).
\]  

(18)
Proof. For $i = 1, \ldots, s + 1$,

$$G(K_i) - G(K_{i-1}) \geq \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} G(\tilde{K}) - G(K_{i-1}),$$

(19)

where $\hat{E}$ is a $\beta$ submodular $\psi(n)$-approximation of the $\beta$-submodular cost function $E$, and $\tilde{K}$ is the optimal solution of $\max \{ G(K) | E(K) \leq \beta E_{\max}(1+\beta)(N_u) \}$. Thus for $i = 1$, we can obtain

$$G(K_1) \geq \frac{\hat{E}(K_1) - \hat{E}(K_0)}{E_{\max}} G(\tilde{K}).$$

(20)

For $i > 1$, we can obtain

$$G(K_i) = G(K_{i-1}) + [G(K_i) - G(K_{i-1})]$$

$$\geq G(K_{i-1}) + \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} (G(\tilde{K}) - G(K_{i-1}))$$

$$= \left( 1 - \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} \right) G(K_{i-1}) + \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} G(\tilde{K})$$

$$\geq \left( 1 - \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} \right) \left[ \left( 1 - \prod_{l=1}^{i} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right) G(\tilde{K}) \right]$$

$$+ \frac{\hat{E}(K_i) - \hat{E}(K_{i-1})}{E_{\max}} G(\tilde{K})$$

$$= \left( 1 - \prod_{l=1}^{i} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right) G(\tilde{K}).$$

(21)

\[ \square \]

Theorem 2. The UCR algorithm can achieve a bi-criterion approximation ratio of $$(1 - 1/e)/2.$$ 

Proof. From Lemma 2, we can obtain

$$G(K_{s+1}) \geq \left[ 1 - \prod_{l=1}^{s+1} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right] G(\tilde{K}).$$

(22)

From the fact that the $s + 1$ iteration violates the UAV energy consumption constraint, we have $\hat{E}(K_{s+1}) > E_{\max}$. Then we can obtain

$$G(K_{s+1}) \geq \left[ 1 - \prod_{l=1}^{s+1} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right] G(\tilde{K}).$$

(23)

For $z_1, \ldots, z_n \in R^+$ such that $\sum z_i = Z, 1 - \prod_{i=1}^{n} (1 - z_i / Z)$ achieves minimum value at $z_1 = \ldots = z_n = Z/n$. Therefore, we have

$$G(K_{s+1}) \geq \left[ 1 - \prod_{l=1}^{s+1} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right] G(\tilde{K})$$

$$\geq \left[ 1 - \left( 1 - \frac{1}{s + 1} \right) \prod_{l=1}^{s+1} \left( 1 - \frac{\hat{E}(K_l) - \hat{E}(K_{l-1})}{E_{\max}} \right) \right] G(\tilde{K})$$

(24)

\[ \square \]
In addition, since we proved that \( G(K) \) is a submodular function in the above, we have \( G(K_{s+1}) - G(K_s) \leq G(K_{s+1}) - G(K^*) \), where \( K^* = \arg \max_{K \in \Psi} G(K) \). Therefore, we can obtain
\[
G(K_s) + G(K^*) \geq G(K_{s+1}) \geq \left( 1 - \frac{1}{e} \right) G(K) \tag{25}
\]
and
\[
\max\{G(K_s), G(K^*)\} \geq \frac{1}{2} \left( 1 - \frac{1}{e} \right) G(K). \tag{26}
\]

To sum up, the UCR algorithm has a bi-criterion approximation ratio of \( (1 - 1/e)/2 \).

5. Simulations and Evaluations

In this section, we simulate the system under different settings and evaluate our algorithm in terms of total shooting utility and time utilization efficiency. The overall shooting utility was introduced in the system model, and the time utilization efficiency can be obtained by dividing the total shooting time of the UAV for the target at the shooting point by the total hovering time, which can be expressed by the formula
\[
\sum_{i=1}^{n} t_{ij} / \sum_{i=1}^{n} t_i.
\]
By default, the task area is 500 m \( \times \) 500 m in size, the numbers of ground targets and cluster heads are 40 and 10, respectively, and the maximum shooting utility per target is 220. Some other important parameters of the simulation are shown in Table 2 [27,28]. In each different parameter setting, we conducted 1000 simulations and took the average value to obtain the following experimental results.

Table 2. Simulation Parameters.

| Symbol | Definition                  | Value          |
|--------|----------------------------|----------------|
| \( m \) | Mass of UAV               | 2.07 kg        |
| \( E_{max} \) | Energy capacity of UAV | \( 4 \times 10^4 \) J |
| \( h \) | Altitude of UAV           | 30 m           |
| \( v_a \) | Velocity of UAV          | 20 m/s         |
| \( r_u \) | Rotor radius              | 0.127 m        |
| \( n_u \) | Number of rotors         | 4              |
| \( \rho \) | Air density               | 1.225 kg/m^3   |
| \( R \) | Shooting radius of UAV    | 16 m           |

In order to evaluate the performance of our proposed UCR algorithm, we compare it with the following three algorithms:

- Random algorithm (RAN): Within the energy consumption constraint, the UAV randomly selects cluster heads for shooting.
- Maximum shooting utility point selection algorithm (MSU): The essence of the MSU algorithm is to select the cluster head with the greatest shooting utility in each iteration. This process is repeated until the UAV’s energy consumption exceeds the UAV’s energy capacity or all cluster heads have been selected.
- Optimal algorithm (OPT): We use the powerful optimization software LINGO to obtain the optimal solution to analyze the gap between our proposed algorithm and the optimal one. Because of the high computational complexity, we run it once for different settings to obtain the optimal solution, and we can only use it to solve small instances of this problem.

5.1. Impact of the Number of Targets

In Figure 3, the number of ground targets varies from 40 to 100, and we analyze the effects of different numbers of ground targets on shooting utility and time utilization efficiency. It can be seen from Figure 3a that when the number of ground targets is 40, the performance of the UCR algorithm is almost close to the optimal scheme. With an increase in the number of ground targets, the gap between the performance of the UCR algorithm and the OPT algorithm increases from 7.0% to 16.3%. Nevertheless, our proposed
algorithm still outperforms the RAN and MSU algorithms in the case of a large number of ground targets. Compared to the RAN algorithm, our algorithm improves shooting utility by 25–50.1%, which is reasonable because RAN randomly selects cluster heads while our algorithm selects cluster heads with the maximum utility-cost ratio at each iteration, taking into account the energy consumption of the UAV, thus saving motion energy and improving shooting utility. Compared to the MSU algorithm, the performance of this algorithm is improved by 14.9–21.9%. This is reasonable because the MSU algorithm only selects the cluster head with the greatest shooting utility in each iteration, while ignoring the energy consumption of the UAV, which leads to higher energy consumption. As shown in Figure 3b, the utilization rate of the hover time of the UAV gradually increases with an increase in ground targets, which is reasonable because when there is a sufficient number of ground targets, shooting utility no longer increases significantly. When the number of ground targets is 100, the optimal scheme can achieve a time utilization rate of 90.8%, and our algorithm can achieve a time utilization rate of 84.4%, which is better than 74.1% for MSU and 52.5% for RAN.

Figure 3. Impact of the number of targets. (a) Overall shooting utility; (b) Time utilization efficiency.

5.2. Impact of the Radius of Clusters

Figure 4 shows the trend of shooting utility and time utilization efficiency with the radius of clusters. We set the number of ground targets to 40. It can be seen from Figure 4a that with an increase in the radius of the clusters, shooting utility gradually decreases. We note that the average shooting utility of our algorithm is 28.5% higher than that of RAN and 24% higher than that of MSU. In order to compare the gap between our algorithm and the OPT algorithm more intuitively, we give approximations of the shooting utility of the two algorithms for different cluster radius parameters in Table 3. In Figure 4b, we can see that the time utilization shows little change with an increase in the radius of clusters and shows a downward trend as a whole. At the same time, we can still see that the UCR algorithm is the closest to OPT. When the radius of clusters is 10 m, our algorithm achieves a time utilization rate of 80.3%, which is higher than 69.5% for MSU and 60.6% for RAN.

Table 3. Approximation of Shooting Utility Between UCR and OPT Algorithms Under Different Cluster Radius Parameters.

| Radius of clusters (m) | 10  | 11  | 12  | 13  | 14  | 15  | 16  |
|------------------------|-----|-----|-----|-----|-----|-----|-----|
| Approximation of shooting utility | 94.76% | 94.91% | 95.05% | 94.69% | 95.13% | 95.19% | 95.58% |
5.3. Impact of the UAV Energy Capacity

In Figure 5, the UAV energy capacity changes from \(10^4\) J to \(7 \times 10^4\) J. Figure 5a shows that when the energy capacity of the UAV increases, the target shooting utility of the four algorithms is improved. Moreover, the overall shooting utility UCR outperforms MSU and RAN on average by 11.2% and 57.3%, respectively. This is reasonable because, with the growth of the UAV’s energy capacity, more cluster heads can be selected for shooting. As shown in Figure 5b, with the increase in the maximum energy of the UAV, the time utilization efficiency shows an upward trend. It can be seen that when the energy capacity of the UAV is \(5 \times 10^4\) J, the UCR algorithm can achieve a time utilization rate of 74.9%, which is close to the 83.2% of the OPT algorithm and better than 66.5% for MSU and 44% for RAN.

5.4. Impact of the Target Distribution Density

Figure 6 shows the trend of target shooting utility and time utilization efficiency with the target distribution density. Assuming that the mission area is square, we set the number of ground targets to 60 and change the side length of the area from 400 m to 1000 m. In Figure 6a, we can see that as the side length of the mission area increases, shooting utility shows a downward trend. This is reasonable because when the number of targets is fixed, a larger mission area means a greater distance between target points and, therefore, a more scattered ground target distribution, which results in more energy consumed for UAV movement and less energy for shooting. As shown in Figure 6a, the UCR algorithm shows better performance with respect to shooting utility. For example, when the side length of the mission area is 400 m, the shooting utility of this algorithm is improved by 20% and 41.1%, respectively, compared with the MSU and RAN algorithms. As shown in Figure 6b,
the time utilization shows a decreasing trend as the task area expands. This is reasonable because when the mission area increases, the distribution of ground targets is scattered, and there are fewer targets in each cluster, so the shooting time of the UAV when hovering at the cluster head decreases. When the side length of the mission area is 400 m, our algorithm achieves a time utilization rate of 88.4%, which is higher than 80.1% for MSU and 51.5% for RAN.

![Figure 6](image1)

**Figure 6.** Impact of the target distribution density. (a) Overall shooting utility; (b) Time utilization efficiency.

### 5.5. Impact of the UAV Parameters

The parameters of the UAV have an impact on its motion energy consumption and hovering energy consumption. For example, differences in the weight, flight speed and number of rotors of the UAV yield different results. As shown in Figure 7, we set $\lambda$ from 4 to 10 and $\mu$ from 1 to 4 to evaluate the effect on the overall shooting utility. We can see that smaller values for $\lambda$ and $\mu$ result in higher shooting utility, which is reasonable because $\lambda$ and $\mu$ represent the energy consumption per unit distance of UAV flight and the energy consumption per unit time of UAV hovering, respectively. Therefore, lower values for these variables result in a reduction in energy consumption and an increase in shooting utility.

![Figure 7](image2)

**Figure 7.** Overall shooting utility versus parameters $\lambda$ and $\mu$.

### 5.6. Trajectory Planning of the UAV

In this section, we use Figure 8 to represent the UAV trajectory generated by each algorithm. From the figure, we see that our algorithm considers the energy consumption of
the UAV in each iteration, unlike the MSU and RAN algorithms, so more cluster heads are selected, and the trajectory obtained by our algorithm does not produce a loop. In addition, we can also see that the cluster heads chosen by our algorithm are the same as those of the OPT algorithm.

Figure 8. Trajectory planning of the UAV. (a) OPT algorithm; (b) UCR algorithm; (c) MSU algorithm; (d) RAN algorithm.

5.7. Algorithm Comparison

In Table 4, we compare the advantages of the four algorithms in terms of time complexity, global optimal solution and applicable scale.

6. Conclusions

In this paper, we considered a UAV-assisted shooting mission scenario. In order to improve the utility of shooting targets, we optimized UAV trajectory planning, hovering time and shooting point selection. By proving the submodularity of the utility function, we transformed the initial problem into a problem of maximizing the submodularity function under path constraints. We proposed an approximate algorithm to maximize shooting utility through two-level optimization, and the approximate ratio of the algorithm is proved to be \((1 - 1/e)/2\). Simulation results showed that the performance of our algorithm is closest to the optimal algorithm, and compared with the MSU and RAN algorithms, our proposed algorithm improves the overall shooting utility and time utilization efficiency. In this paper, we only consider the scenario where a single UAV completes the task, but multi-
UAV cooperation can achieve more efficient shooting tasks. In the future work, we will consider the communication between UAVs and dynamically realize the task scheduling and trajectory planning of distributed UAVs by using deep reinforcement learning.

**Author Contributions:** Conceptualization, C.D. and W.W.; formal analysis, Y.W. and W.W.; visualization, Y.W.; supervision, S.F. and C.D.; writing—original draft, Y.W.; writing—review and editing, S.F. and W.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded in part by the Key Research and Development project of Jiangsu Province under Grant BE2021013-4; in part by China NSF under Grant 61931011; in part by China NSF under Grant 62001219; and in part by the Shuangchuang Talent Program of Jiangsu Province under Grant JSSCBS20210159.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The researchers would like to thank the funding support.

**Conflicts of Interest:** The researchers declare no conflict of interest.

**Appendix A**

According to the definition of the shooting utility function, shooting utility increases with an increase in the number of cluster heads. Thus, for any $A \subseteq B \subseteq K$, $G_A(o_j) \leq G_B(o_j)$. Due to the fact that the total shooting utility is superimposed, we obtain the following relationship:

$$G(A) = \sum_{j=1}^{m} G_A(o_j) \leq \sum_{j=1}^{m} G_B(o_j) = G(B),$$  \hspace{1cm} (A1)

which implies that $G(K)$ is monotone.

From the definition of the substitution function, $G(K) = Q(K, T') \geq 0$, thus it is nonnegative.

For submodularity, we prove it by proving the following inequation $G(A + c_i) - G(A) \geq G(B + c_i) - G(B)$. It is equivalent to proving that for each target $o_j$, $G_{A+c_i}(o_j) \geq G_B(o_j)$ for $G(A \cup \{c_i\})$ to make the expression more intuitive.

It can be seen from the superposition of the shooting utility, $q_A(o_j) \leq q_B(o_j)$, then we can divide the proof problem into the following three cases.

**Case 1.** $q_A(o_j) \leq q_B(o_j) \leq Q^m_j$: In this case,

$$G_{A+c_i}(o_j) - G_A(o_j) = \min\left\{q_{A+c_i}(o_j), Q^m_j\right\} - \min\left\{q_A(o_j), Q^m_j\right\}$$

$$= \min\left\{q(o_j, c_i), Q^m_j - q_A(o_j)\right\}$$  \hspace{1cm} (A2)

and

$$G_{B+c_i}(o_j) - G_B(o_j) = \min\left\{q_{B+c_i}(o_j), Q^m_j\right\} - \min\left\{q_B(o_j), Q^m_j\right\}$$

$$= \min\left\{q(o_j, c_i), Q^m_j - q_B(o_j)\right\}.$$  \hspace{1cm} (A3)

Since $Q^m_j - q_B(o_j) \leq Q^m_j - q_A(o_j)$, we can consider the following three cases to prove the inequality.

(a) If $q(o_j, c_i) \leq Q^m_j - q_B(o_j) \leq Q^m_j - q_A(o_j)$, we have

$$G_{A+c_i}(o_j) - G_A(o_j) = q(o_j, c_i) = G_{B+c_i}(o_j) - G_B(o_j).$$  \hspace{1cm} (A4)
(b) If $Q^m_j - q_B(o_j) < q(o_j, c_i) < Q^m_j - q_A(o_j)$, we have

$$G_{A+c_i}(o_j) - G_A(o_j) = q(o_j, c_i) > Q^m_j - q_B(o_j) = G_{B+c_i}(o_j) - G_B(o_j).$$  \hspace{1cm} (A5)

(c) If $Q^m_j - q_B(o_j) \leq Q^m_j - q_A(o_j) \leq q(o_j, c_i)$, we have

$$G_{A+c_i}(o_j) - G_A(o_j) = Q^m_j - q_A(o_j) \geq Q^m_j - q_B(o_j) = G_{B+c_i}(o_j) - G_B(o_j).$$  \hspace{1cm} (A6)

Case 2. $q_A(o_j) \leq Q^m_j \leq q_B(o_j)$: In this case,

$$G_{A+c_i}(o_j) - G_A(o_j) = \min \{q_{A+c_i}(o_j), Q^m_j\} - \min \{q_A(o_j), Q^m_j\}$$

$$= \min \{q_A(o_j), Q^m_j\} - q_A(o_j) \geq 0,$$

$$G_{B+c_i}(o_j) - G_B(o_j) = \min \{q_{B+c_i}(o_j), Q^m_j\} - \min \{q_B(o_j), Q^m_j\}$$

$$= Q^m_j - Q^m_j = 0.$$  \hspace{1cm} (A7)

Case 3. $Q^m_j \leq q_A(o_j) \leq q_B(o_j)$: In this case,

$$G_{A+c_i}(o_j) - G_A(o_j) = \min \{q_{A+c_i}(o_j), Q^m_j\} - \min \{q_A(o_j), Q^m_j\}$$

$$= Q^m_j - Q^m_j = 0,$$

$$G_{B+c_i}(o_j) - G_B(o_j) = \min \{q_{B+c_i}(o_j), Q^m_j\} - \min \{q_B(o_j), Q^m_j\}$$

$$= Q^m_j - Q^m_j = 0.$$  \hspace{1cm} (A8)

For the above three cases, $G_{A+c_i}(o_j) - G_A(o_j) \geq G_{B+c_i}(o_j) - G_B(o_j)$ is satisfied, thus we prove that $G(K)$ is a submodular function.

References

1. Wu, D.; Ci, S.; Luo, H.; Ye, Y.; Wang, H. Video Surveillance Over Wireless Sensor and Actuator Networks Using Active Cameras. IEEE Trans. Autom. 2011, 56, 2467–2472. [CrossRef]  [CrossRef]
2. Abbas, N.; Yu, F. Design and Implementation of a Video Surveillance System for Linear Wireless Multimedia Sensor Networks. In Proceedings of the 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), Chongqing, China, 27–29 June 2018. [CrossRef]
3. Hsiao, Y.-P.; Shih, K.-P.; Chen, Y.-D. On Full-View Area Coverage by Rotatable Cameras in Wireless Camera Sensor Networks. In Proceedings of the 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), Taipei, China, 27–29 March 2017.
4. Song, K.-T.; Tai, J.-C. Dynamic Calibration of Pan–Tilt–Zoom Cameras for Traffic Monitoring. IEEE Trans. Syst. Man Cybern. 2006, 36, 1091–1103. [CrossRef] [CrossRef] [PubMed]
5. Kanhere, N.K.; Birchfield, S.T. A Taxonomy and Analysis of Camera Calibration Methods for Traffic Monitoring Applications. IEEE Trans. Intell. Transp. Syst. 2010, 11, 441–452. [CrossRef] [CrossRef]
6. Yu, H.; Yang, W.; Zhang, H.; He, W. A UAV-based crack inspection system for concrete bridge monitoring. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017. [CrossRef]
7. DeMoraes, R.S.; DeFreitas, E.P. Multi-UAV Based Crowd Monitoring System. IEEE Trans. Aeros. Electron. Syst. 2020, 56, 1332–1345. [CrossRef] [CrossRef]
8. Hayat, S.; Yanmaz, E.; Muzaffar, R. Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint. IEEE Commun. Surv. Tutor. 2016, 18, 2624–2661. [CrossRef] [CrossRef]
9. Kim, Y.; Min, Y.G.; Hee, P.S.; Wun-Cheol, J.; Soonyong, S.; Tae-Wook, H. The analysis of image acquisition method for Anti-UAV surveillance using cameras image. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 21–23 October 2020. [CrossRef] [CrossRef]
10. Sanchez-Fernandez, A.J.; Romero, L.F.; Bandera, G.; Tabik, S. VPP: Visibility-Based Path Planning Heuristic for Monitoring Large Regions of Complex Terrain Using a UAV Onboard Camera. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2022, 15, 944–955. [CrossRef] [CrossRef]
11. Vazquez-Carmona, E.V.; Vazquez-Gomez, J.I.; Herrera-Lozada, J.C. Environmental Monitoring using Embedded Systems on UAV. *IEEE Lat. Am. Trans.* 2020, 18, 303–310. [CrossRef] [CrossRef]
12. Gao, J.; Hu, Z.; Bian, K.; Mao, X.; Song, L. AQ360: UAV-Aided Air Quality Monitoring by 360-Degree Aerial Panoramic Images in Urban Areas. *IEEE Internet Things J.* 2021, 8, 428–442. [CrossRef] [CrossRef]
13. Ji, Y.; Dong, C.; Wu, Q. Fair-energy trajectory plan for multi-target omnidirectional coverage based on cooperative UAVs. In Proceedings of the IET 8th International Conference on Wireless, Mobile Multimedia Networks(ICWMMN), Beijing, China, 15–17 November 2019. [CrossRef]
14. Zhan, C.; Zeng, Y.; Zhang, R. Energy-Efficient Data Collection in UAV Enabled Wireless Sensor Network. *IEEE Wirel. Commun. Lett.* 2018, 7, 328–331. [CrossRef] [CrossRef]
15. Wang, W.; Peng, Y.; Cao, G.; Guo, X.; Kwok, N. Low-Illumination Image Enhancement for Night-Time UAV Pedestrian Detection. *IEEE Trans. Industr. Inform.* 2021, 17, 5208–5217. [CrossRef] [CrossRef]
16. Yuan, Y.; Fang, F.; Zhang, G. Superpixel-Based Seamless Image Stitching for UAV Images. *IEEE Trans. Geosci. Remote Sens.* 2021, 59, 1565–1576. [CrossRef] [CrossRef]
17. Zhang, F.; Yang, T.; Liu, L.; Liang, B.; Bai, Y.; Li, J. Image-Only Real-Time Incremental UAV Image Mosaic for Multi-Strip Flight. *IEEE Trans. Multimed.* 2021, 23, 1410–1425. [CrossRef] [CrossRef]
18. Wang, L.-J.; Hsieh, W.-S. Toward an Improvement of UAV-Aerial Image Using Non-linear Image Enhancement. In Proceedings of the 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), Krakow, Poland, 16–18 May 2018. [CrossRef]
19. Li, Y.; Liang, W.; Xu, W.; Xu, Z.; Jia, X.; Xu, Y.; Kan, H. Data Collection Maximization in IoT-Sensor Networks Via an Energy-Constrained UAV. *IEEE Trans. Mob. Comput.* 2021. [CrossRef] [CrossRef]
20. Chen, M.; Liang, W.; Das, S.K. Data Collection Utility Maximization in Wireless Sensor Networks via Efficient Determination of UAV Hovering Locations. In Proceedings of the 2021 IEEE International Conference on Pervasive Computing and Communications (PerCom), Kassel, Germany, 22–26 March 2021. [CrossRef]
21. Youssef, M.; Youssef, A.; Younis, M. Overlapping Multihop Clustering for Wireless Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* 2009, 20, 1844–1856. [CrossRef] [CrossRef]
22. Baadel, S.; Thabtah, F.; Lu, J. Overlapping clustering: A review. In Proceedings of the 2016 SAI Computing Conference (SAI), London, UK, 13–15 July 2016. [CrossRef]
23. Wang, W.; Dai, H.; Dong, C.; Xiao, F.; Zheng, J.; Cheng, X.; Chen, G.; Fu, X. Deployment of Unmanned Aerial Vehicles for Anisotropic Monitoring Tasks. *IEEE Trans. Mob. Comput.* 2022, 21, 495–513. [CrossRef] [CrossRef]
24. Zheng, Z.; Shroff, N.B. Submodular Utility Maximization for Deadline Constrained Data Collection in Sensor Networks. *IEEE Trans. Autom.* 2014, 59, 2400–2412. [CrossRef] [CrossRef]
25. Qin, Z.; Dong, C.; Wang, H.; Li, A.; Dai, H.; Sun, W.; Xu, Z. Trajectory Planning for Data Collection of Energy-Constrained Heterogeneous UAVs. *Sensors* 2019, 19, 4884. [CrossRef] [CrossRef] [PubMed]
26. Zhang, H.; Vorobeychik, Y. Submodular optimization with routing constraints. In Proceedings of the Thirtieth Aaai Conference on Artificial Intelligence (AAAI), Phoenix, AZ, USA, 12–17 February 2016.
27. Monwar, M.; Semiari, O.; Saad, W. Optimized Path Planning for Inspection by Unmanned Aerial Vehicles Swarm with Energy Constraints. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018. [CrossRef]
28. Chiping, T.; McKinley, P.K. Energy Optimization under Informed Mobility. *IEEE Trans. Parallel Distrib. Syst.* 2006, 17, 947–962. [CrossRef] [CrossRef]