Application of distributed auction to multi-UAV task assignment in agriculture

Jie Hu*, Jiacheng Yang

(*National Center for International Collaboration Research on Precision Agricultural Aviation Pesticides Spraying Technology/ College of Electronic Engineering, South China Agricultural University, Guangzhou 510642, China*)

Abstract: This paper addresses the multiple farming task assignment for agricultural UAVs by presenting a decentralized auction algorithm, where computation and information are distributed among multiple parallel processing units (UAVs). The scheme iterates between bundle construction phase and two conflict resolution phases, and then converges to a task allocation and route plan simultaneously. In the bundle construction stage, each UAV groups the tasks with commonalities by considering its own capacities of flight endurance, weight load, battery, data storage, etc. In the following conflict resolution stages, the winning UAVs for tasks are determined by the information exchanged between UAVs. Later, the proposed algorithm is shown to have a low demand for communication and is proved to be able to achieve 50% optimality. Finally, an application of collecting the data of ground sensor nodes (including moving nodes) is used to assess the performance of the proposed scheme. Numerical experiments confirm superior convergence properties and performance of the proposed algorithm when compared with existing task-allocation algorithms.

Keywords: unmanned aerial vehicle, farming task allocation, route plan, distributed, auction, agricultural aviation

DOI: 10.33440/j.ijpaa.20180101.0008

Citation: Hu J, Yang J C. Application of distributed auction to multi-UAV task assignment in agriculture. Int J Precis Agric Aviat, 2018; 1(1): 44–50.

1 Introduction

Unmanned aerial vehicles have been developed and applied to perform farming tasks such as remote sensing[1], crop dusting[2], pesticide spraying[3,4], routine monitoring of crop plant health[5] and acquiring sensed data from ground sensor nodes[6-8]. Nevertheless, available agricultural UAVs are still limited by their payload and flight endurance. For example, most mini fixed-wing airplanes or rotary-winged helicopters used for low altitude remote sensing in agriculture have short endurance[9], which is always less than an hour[10]. For a large area over hundreds of hectares, a UAV will take a long time to traverse all predefined waypoints and has to return back for new power from time to time. Even for a larger payload capacity and longer flight endurance spraying UAV, limited pesticide loads also impose restrictions on its widely use. To avoid the unnecessary time waste in turning-around for power and reduce the total task completion time, more UAVs may be needed to cover a large farm field.

Although we can take an advantage of the high efficiency of multi UAVs in doing agricultural tasks, the complexity for global computation and information of the task assignment problem among multiple UAVs can be very high. One difficulty lies in the dependency between task assignment and path planning. Due to the limitation in payload and flight endurance of agricultural UAVs, the cost to complete a task is dependent on the path. Thus, route plan should be considered when tasks are assigned. The general way to solve the task allocation and path planning problem is allocate-then-plan schemes, which assigns the tasks first, then plans the route for each UAV. However, the allocate-then-plan schemes decouple the allocation and route planning problems, do not consider the complete solution space and may find inefficient solutions.

In order to deal with task allocation and route planning simultaneously, most techniques require a robust central processor that handles all computation and information in the system. However in agriculture, a robust central processor may not always be available at the field site, thus gives rise to the need for decentralized solutions, where computation and information are distributed among multiple parallel processing units (corresponding to the UAVs). One such approach is the auction algorithm which allows UAVs to bid for the tasks according to their states and capabilities[1].

In auction, the traditional way of computing the winner is to have a central system acting as the auctioneer to receive and evaluate each bid among the UAVs[11]. A winner is selected after all of the bids are collected by the auctioneer. Some approaches choose one of the bidders to act as the auctioneer[12]. Since the UAVs may fly away from the predefined auctioneer, especially when doing an agricultural mission in a large farm field, the above approaches can't be used for mobile bidders such like UAVs because not all of the UAVs can send their bids to the auctioneer directly. We address this challenge by means of a distributed auction algorithm, where the UAVs are able to bid for the tasks according to their local information and resolve the conflict by themselves, no central processor or auctioneer are needed.

In agriculture, each UAV has to take more than one task along its flight route. As the cost or the score for a task is dependent on the route, thus the cost or the score for the task is also dependent on the other tasks this UAV takes. The dependency between tasks

Received date: 2018-02-11 **Accepted date:** 2018-05-28

Biographies: Jiacheng Yang, Master student, research interests: agricultural informatization, precision agricultural aviation application, Email: 532304223@qq.com.

***Corresponding author: Jie Hu**, PhD, Associate professor, research interests: agricultural informatization, precision agricultural aviation applications. College of Electronic Engineering, South China Agricultural University, Guangzhou 510642, China. Tel: +86-13822123081, Email: hjgz79@scau.edu.cn.

makes the task assignment problem more complicated. In many cases, this is done by running sequential auctions and awarding a single task at a time until there are no remaining tasks left to assign[13]. However, sequential auction sacrifices the total benefit because it lacks global view. In order to improve the total benefit, combinational auction was developed to allow UAVs to bid on task groups rather than the individual tasks. Nevertheless, difficulties can arise in the computational cost for enumerating all possible task combinations and determining the winner among these combinations. The winner determination has been shown to be NP (Nondeterminism Polynomial) hard problem.

Another property of task allocation problem among agricultural UAVs is the limited capabilities of UAVs due to its flight endurance and weight load. This limitation should be considered in allocation, especially when taking tasks in a large farmland.

In this paper, we developed a distributed auction to solve the multi-task allocation problem among multi UAVs in agriculture. Based on the auction algorithms as in [14-16], our approach groups tasks that have commonalities, enables multiple tasks to be assigned to a UAV and multiple conflicts to be resolved in parallel, thus enormously accelerates the convergence time of multi-task allocation. At the same time, the dependency between agricultural tasks and the capabilities of agricultural UAVs are taken into account when each UAV groups similar tasks. It is worthy to note that the proposed algorithm could perform the tasks allocation and route plan simultaneously, thus achieve better performance than those allocate-then-plan schemes.

2 Problem formation

Assume there are N UAVs with integrated wireless communication capabilities and M tasks. The tasks can be the predefined remote sensing waypoints, the ground sensor nodes with data to be collected or the land parcels where pesticides need to be sprayed. The global objective function is assumed to maximize the sum of rewards of performing the tasks, or to minimize the total costs of executing the tasks. The minimization function can be used for the situation like minimizing total travel distance or minimizing tasks completion time. As maximization problem and minimization problem can be mutually converted, we just consider the maximization problem in this paper. We use u_{ij} to denote the reward when UAV i executing task j. It is also worthy to note that u_{ij} always relates with path route $\boldsymbol{p}_i$ (the flight route of UAV i). In order to avoid UAVs collision, it assumes that each task is assigned to no more than one UAV. The task assignment problem then can be written as the following integer program

$$\max_{x_{ij},\boldsymbol{p}_i} \sum_{i=1}^{N}\sum_{j=1}^{M} u_{ij}(\boldsymbol{p}_i)x_{ij} \tag{1}$$

Subject to

$$\sum_{i=1}^{N} x_{ij} \le 1 \quad \forall j \in \boldsymbol{J} \tag{2a}$$

$$\sum_{j=1}^{M} c_{ij}(\boldsymbol{p}_i)x_{ij} \le C_i \quad \forall i \in \boldsymbol{I} \tag{2b}$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J} \tag{2c}$$

where, x_{ij}=1 means task j is assigned to UAV i, $\boldsymbol{I}$ is the UAVs' set and $\boldsymbol{J}$ is the tasks' set. Formula (2a) means one task can't be allocated to more than one UAV. Formula (2b) means each UAV is limited by its available resources (endurance, fuel or battery consumption, pesticide loads, memory storage capacity, etc.), where $c_{ij}(\boldsymbol{p}_i)$ denotes the resource consumption for UAV i to perform task j along the path route $\boldsymbol{p}_i$, and C_i denotes the capability of UAV i.

Let $S_i^{\boldsymbol{p}_i}$ be defined as the total reward for UAV i performing its tasks along the path $\boldsymbol{p}_i$, then

$$u_{ij}(\boldsymbol{p}_i) = \max_{m \le |\boldsymbol{p}_i|} S_i^{\boldsymbol{p}_i \oplus_m \{j\}} - S_i^{\boldsymbol{p}_i} \tag{3}$$

where, $\oplus_m$ means inserting the right list after the mth element of the left list, and | | denotes the cardinality of a list. Equation (3) expresses how a UAV estimate the reward of a task. When adding a new task to its task set, the UAV will insert it in an optimal position along the flight route which maximizing its total reward. In the paper, we assume that the reward of performing a task is characterized by a non-decreasing, submodular function[17,18]. Specially, it satisfies the follow conditions

$$\begin{cases} \sum_{j=1}^{M} u_{ij}(\boldsymbol{p}_i) \le \sum_{j=1}^{M} u_{ij}(\boldsymbol{p}_i \oplus_m \{k\}) \\ u_{ij}(\boldsymbol{p}_i) \le u_{ij}(\boldsymbol{p}_i \oplus_m \{k\}), \quad \forall j \in \boldsymbol{p}_i \end{cases} \tag{4}$$

Note that if more tasks are assigned to UAV i, it will gain more utility. However, if a new task k is added to the task set, the reward of any original task does not increase. Though not all of the utility functions of interest in task allocation satisfy submodular, but in agriculture, this assumption on utility function is general enough to capture the majority applications [14]. For example, when a UAV take remote sensing for a particular area, with the pictures and data increasing, the incremental utility may be tiny by adding new sensing pictures at the same block.

3 Distributed auction algorithm

The distributed auction algorithm for multi-task allocation among multi-UAV includes three phases. The first phase is task combination (bundle) construction and route planning, this phase is executed locally by each UAV. In the second and third phase, UAVs communicate to resolve conflicts.

A. Phase 1

Phase 1 describes how the task combination (bundle) $\boldsymbol{b}$ and path route $\boldsymbol{p}$ are locally built by each UAV. A UAV compares the reward for adding a new task into the current bundle and current path with the current winning bid list's $\boldsymbol{y}$ value for that task. If the reward is greater than the current winning bid, the UAV assigns itself the task.

Table 1 Phase 1 of the distributed auction algorithm for multi-task allocation among multi-UAV

Phase 1: Build bundle and plan route for UAV i
$\boldsymbol{y}_i(t)=\boldsymbol{y}_i(t-1)$ $\boldsymbol{p}_i(t)=\boldsymbol{p}_i(t-1)$ $\boldsymbol{b}_i(t)=\boldsymbol{b}_i(t-1)$ $\boldsymbol{z}_i(t)=\boldsymbol{z}_i(t-1)$
while $c_i<C_i$ and $h_i \neq 0$ do
$u_{ij}=\max_{m\le\lvert p_i\rvert} S_i^{p_i\oplus_m\{j\}}-S_i^{p_i}, \forall j\in \boldsymbol{J}\setminus \boldsymbol{b}_i$
if $u_{ij}>y_{ij}$ h_{ij}=1, else h_{ij}=0, $\forall j\in \boldsymbol{J}$
$J_i=\arg\max_{j} u_{ij}\cdot h_{ij}$
$m_{i,J_i}=\arg\max_{m} S_i^{p_i\oplus_m\{J_i\}}$
$\boldsymbol{p}_i=\boldsymbol{p}_i\oplus_{m_{i,J_i}}\{J_i\}$ $\boldsymbol{b}_i=\boldsymbol{b}_i\oplus_{end}\{J_i\}$ $c_i=F(\boldsymbol{p}_i,\boldsymbol{b}_i)$
$y_{i,J_i}(t)=u_{i,J_i}$
$z_{i,J_i}(t)=i$
end while

Where the bundle $\boldsymbol{b}_i$ lists the tasks in the order which they were added sequentially, whereas, the path $\boldsymbol{p}_i$ is the order in which the tasks are performed. Except for $\boldsymbol{b}_i$ and $\boldsymbol{p}_i$, each UAV carries the other two vectors: a winning bid list $\boldsymbol{y}_i$ and winning UAVs list $\boldsymbol{z}_i$. $F(\boldsymbol{p}_i, \boldsymbol{b}_i)$ is the function to calculate the resource consumption of UAV when completing current bundle $\boldsymbol{b}_i$ along the path route $\boldsymbol{p}_i$. In agriculture, this could be the fuel consumption, battery consumption, pesticide consumption, etc. The resource capacity of UAV i is assumed to be C_i. During phase 1, each UAV continuously adds tasks to its bundle until it is incapable of adding any other task. By logically grouping tasks with commonalities, our algorithm may achieve faster converge speed than its sequential counterparts and have better performance in the assignment.

B. Phase 2

Table 2 Phase 2 of the distributed auction algorithm for multi-task allocation among multi-UAV

Phase 2: conflict resolution Ⅰ for UAV i

1: send $\boldsymbol{y}_i$ and z_i *to* other UAV $k(\forall k \in \boldsymbol{I} \setminus i)$
2: receive $\boldsymbol{y}_k$ and z_k from other UAV $k(\forall k \in \boldsymbol{I} \setminus i)$
3: for $j = 1:M$
4: $y_{i,j} = \max y_{k,j}, \quad \forall k \in \boldsymbol{I}$
5: $z_{i,j} = \arg\max_k y_{k,j}, \forall k \in \boldsymbol{I}$
6: end
7: $\overline{n}_i = \min\{n : z_{i,b_{i,n}} \neq i\}$
8: for $\forall n > \overline{n}_i$ and $n \in \boldsymbol{b}_i$
9: if $z_{i,b_{i,n}} == i$
10: $y_{i,b_{i,n}} = 0$
11: $z_{i,b_{i,n}} = \varnothing$
12: end
13: end
14: $b_{i,n} = \varnothing, \quad \forall n \geq \overline{n}_i$

In phase 2, UAVs communicate to resolve conflicts. After updating the winning UAV list z_i and current task price $\boldsymbol{y}_i$ (line 3-6), each UAV i loses its assignment if it finds that it is outbid by other UAVs for the task it had selected. Line 7 is for UAV i to find the first task in its bundle which was thought to assign to itself but is outbid by other UAVs. Where $b_{i,n}$ denotes the nth entry of bundle $\boldsymbol{b}_i$. The reason to reset the tasks and the corresponding winning bids and winning UAVs (line 8-14) which were added after $b_{i,n}$ is that, in phase 1, UAVs add tasks to their bundle based on their current assigned task set, if a UAV is outbid for a task and, thus, releases it, then the estimated reward for the tasks added to the bundle after this task are no longer valid. Accordingly, all of the following winning tasks of UAV i need to be reset (Line 8-13) and all of the following tasks need to be released (line 14).

C. Phase 3

Because some tasks may be released in phase 2, the bids list $\boldsymbol{y}_i$, which is sent by UAV i at the beginning of phase 2, may be invalid. Suppose a UAV is able to release tasks without another UAV selecting it (the invalid bids may prevent other UAVs to choose those tasks), a simple application of the maximum bids update on the winning bids list $\boldsymbol{y}_i$ will no longer converge to the appropriate values since then, the maximum bid observed might no longer be valid. Therefore, Phase 3 is needed. Line 3-8 is to find the invalid bids and reset them as zero. Line 5 means UAV i think the winning UAV for task j is UAV k, but UAV k has released and reset task j in phase 2, thus the invalid winning bids $y_{i,j}$ should be reset. Finally, all winning bids and winning UAVs are updated according to the most current information.

After executing the third phase, the algorithm goes back to the first phase, and new tasks are added. The algorithm iterates between Phases 1-3 until consensus is achieved, which means, for any UAV $i \in I$, $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ do not change over time.

Table 3 Phase 3 of the distributed auction algorithm for multi-task allocation among multi-UAV

Phase 3: Conflict Resolution Ⅱ for UAV i

1: send $\boldsymbol{y}_i$ and z_i *to* other UAV $k(\forall k \in \boldsymbol{I} \setminus i)$
2: receive $\boldsymbol{y}_k$ and z_k from other UAV $k(\forall k \in \boldsymbol{I} \setminus i)$
3: for $j = 1:M$
4: if $z_{k,j} == \varnothing, \quad \forall k \in I \setminus i$
5: if $z_{i,j} == k$
6: $y_{i,j} = 0$
7: end
8: end
9: $y_{i,j} = \max y_{k,j}, \quad \forall k \in \boldsymbol{I}$
10: $z_{i,j} = \arg\max_k y_{k,j}, \forall k \in \boldsymbol{I}$
11: end

It is worthy to note that the algorithm is able to outbid earlier allocated tasks in the conflict resolution stage, which helps provide better assignments. Furthermore, by implementing our auction algorithm, each UAV can not only get tasks assigned, but also complete the route planning to execute the tasks.

Figure 1 shows the communication protocol for the distributed auction. Only two agents are depicted for clarity. At the beginning of each iteration, the most currently information of the tasks is achieved and updated. This process is necessary in case new tasks add in or the states of some tasks change. Note that the loop will continue to execute until leads to a convergence. The convergence property is discussed in the next section.

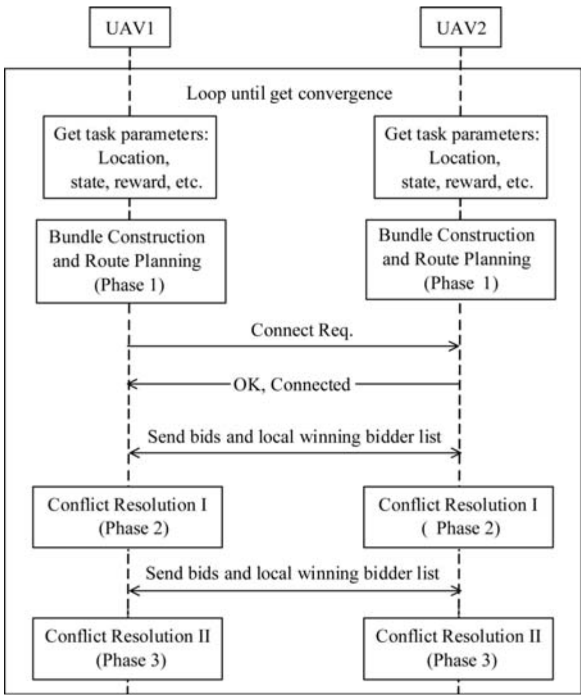

Figure 1 Communication protocol

4 Convergence and performance guarantee

In this section, the Centralized Sequential Greedy Algorithm (CSGA)[15] is introduced in this section to help prove the convergence and performance of our algorithm. The CSGA is shown as follows.

Table 4 Centralized Sequential Greedy Algorithm

Centralized Sequential Greedy Algorithm
$\boldsymbol{p}_i^{(1)} = \{\varnothing\}$, $\boldsymbol{b}_i^{(1)} = \{\varnothing\}$, $\forall i \in I$
for $n = 1 : M$ do
$u_{i,j}^{(n)} = \max_{m \le \lvert \mathrm{p}_i^{(n)} \rvert} S_i^{\mathrm{p}_i^{(n)} \oplus_m \{j\}} - S_i^{\boldsymbol{p}_i^{(n)}}, \forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J}$
$(i_n^*, j_n^*) = \arg\max_{(i,j) \in \mathrm{I} \times \mathrm{J}} u_{i,j}^{(n)}$
$m_{i_n^*, j_n^*} = \arg\max_m S_{i_n^*}^{\mathrm{p}_i \oplus_m \{j_n^*\}}$
$\boldsymbol{p}_{i_n^*}^{(n)} := \boldsymbol{p}_{i_n^*}^{(n)} \oplus_{m_{i_n^*, j_n^*}} \{j_n^*\}$, $\boldsymbol{b}_{i_n^*}^{(n)} := \boldsymbol{b}_{i_n^*}^{(n)} \oplus_{end} \{j_n^*\}$
$\boldsymbol{p}_i^{(n)} = \boldsymbol{p}_i^{(n-1)}, \boldsymbol{b}_i^{(n)} = \boldsymbol{b}_i^{(n-1)}, \quad \forall i \ne i_n^*$
$\boldsymbol{p} = \boldsymbol{p} \setminus \{j_n^*\}$
end for

Note that the superscript (n) denotes the nth step of the algorithm. CSGA sequentially finds a sequence of UAV-task pairs that provide the largest rewards given prior selections. Where $u_{i,j}^{(n)}$ is calculated according to equation (3).

Our algorithm can be proved to converge to the same results as CSGA. The proving method is similar to the convergence proof of CBBA algorithm in [15]. Given the limited space available here，refer to[15] for more details.

Next we will prove CSGA can achieve at least 50% optimality, which means, our distributed auction algorithm can also guarantee 50% optimality as it has the same performance with CSGA.

Lemma 1: The CSGA scheme can achieve 50% optimality.

Proof: We prove it through induction. Let P denote the original problem with N unallocated tasks. We define P' as a new problem with N–1 tasks left, assuming the first task j_1 has already been scheduled by CSGA algorithm. In other words, we can consider CSGA as scheduling task j_1 to UAV i first. Then we run CSGA on problem P' recursively. We use $U_{\mathbf{CSGA}}^{P}$ to denote the reward gained by CSGA algorithm on problem P, and $U_{\mathbf{OPT}}^{P}$ to denote the reward gained by the optimal solution. Clearly, we have the next equation based on the definition of problem P'

$$U_{\mathbf{CSGA}}^{P} = U_{\mathbf{CSGA}}^{P'} + U[\{j_1\}] \tag{4}$$

Next we will show that $U_{\mathbf{CSGA}}^{P} \le U_{\mathbf{CSGA}}^{P'} + 2U[\{j_1\}]$. Let $\boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_3,\ldots\boldsymbol{b}_M$ be the optimal scheduling for problem P, where $\boldsymbol{b}_i$ represents the tasks set allocated to UAV i under optimal scheduling. We next have two cases to study:

Case 1: $j_1 \in \boldsymbol{b}_i$. Since we assume that j_1 is also scheduled to UAV i by CSGA, it indicates that the scheduling of j_1 by CSGA is optimal. Thus, $U_{\mathbf{OPT}}^{P} = U_{\mathbf{OPT}}^{P'} + U[\{j_1\}]$ is true based on the definition of P and P'.

Case 2: $j_1 \notin \boldsymbol{b}_i$. we can modify the optimal schedule by rescheduling j_1 to UAV i. All other tasks still remain to the same scheduling. Obviously, this is a possible allocation for P'. Based on the submodularity of utility function and the greedy manner of CSGA, we can guarantee that the loss resulted from removing j_1 is at most $U[\{j_1\}]$. Thus, we have $U_{\mathbf{OPT}}^{P} - U[\{j_1\}] \le U_{\mathbf{OPT}}^{P'} + U[\{j_1\}]$, then get $U_{\mathbf{OPT}}^{P} \le U_{\mathbf{OPT}}^{P'} + 2U[\{j_1\}]$.

Therefore, $U_{\mathbf{OPT}}^{P} \le U_{\mathbf{OPT}}^{P'} + 2U[\{j_1\}]$ applies for both cases. Finally, the proof can be finished by induction on P'. Thus,

$$U_{\mathbf{OPT}}^{P} \le U_{\mathbf{OPT}}^{P'} + 2U[\{j_1\}] < 2U_{\mathbf{CSGA}}^{P'} + 2U[\{j_1\}] = 2U_{\mathbf{CSGA}}^{P} \tag{5}$$

This finishes the proof.

5 Communication and computational cost issues

In this work, we have assumed that all-to-all communications are available in order for the auction to take place. Actually all-to-all communications are not strictly required, because in the auction rounds it is sufficient to establish a communication ring. Bandwidth restrictions are not significant, as the amount of information exchanged is not a critical issue for our protocol. Considering there are M tasks to be allocated among N UAVs. The parameters need to transmit in phase 2 and phase 3 are $\boldsymbol{y}_i = [y_{i,1}, y_{i,2}, \ldots y_{i,M}], (\forall i \in \boldsymbol{I})$ and $z_i = [z_{i,1}, z_{i,2}, \ldots z_{i,M}], (\forall i \in \boldsymbol{I})$. Each $y_{i,j}$ and $z_{i,j}$ is described by a floating point number. Assuming 4 bytes per number, a total of 4×N×M×2 bytes are sent in phase 2 and phase 3 during each iteration. Note that our auction will converge within M iterations, thus the maximum amount of data need to transmit is $8M^2N$ bytes. As an example, for an assignment of 30 tasks among 5 UAVs, a maximal amount of 36 kilobytes for the whole auction process need to be sent.

As to computational cost, we underline that in a distributed approach such as the one we propose, each UAV just need to carry the computation only for its share. Whereas in a centralized approach, the central processor needs to compute a complete solution, the cost must be multiplied by the number of UAVs or multiplied by the number of tasks.

6 Algorithm performance

In this section we implemented a series of experiments to validate the proposed distributed auction algorithm. A time-discounted reward function as followed was assumed in this section:

$$S_i^{\boldsymbol{p}_i} = \sum \lambda_j^{\tau_i^j(\boldsymbol{p}_i)} c_j \tag{6}$$

where, λ_j<1 was the discounting factor for task j; $\tau_i^j(\boldsymbol{p}_i)$ was the estimated time UAV i took to arrive at task location j along the path $\boldsymbol{p}_i$, and c_j was the reward of performing task j. The time-discounted reward can model the planning of service routes in which satisfaction of client diminishes with time. In agriculture, the time-discounted reward is suitable for many cases. One example is pesticide spraying. The earlier the pesticide can be sprayed, the better the pests can be controlled. Another example is data collecting from ground sensor nodes. Assume the sensor nodes have limited storage but carrying frequently sampling tasks required by some kind of application. Thus after some time's sampling, the data is about to overflow if wasn't collected timely. Therefore, the time-discounted function can properly model the reward of data collecting tasks from ground sensor nodes. In this section, we assumed the sensor nodes were randomly placed on a 8000 m×8000 m field with two classes of sampling requirements. The first class sampling data had an arrival rate of 32 kbps, and the second class sampling data could be mapped to video services, such as the widely known Quarter Common Intermediate Format (QCIF) [19], having an arrival rate of 128 kbps. We assumed all the sensor nodes had their storage being filled up with data, therefore, the new arrived data would overflow if the old data wasn't being collected timely. In this section, the parameters λ=0.95 and c_j≡1 were assumed and the UAVs were considered as having a constant velocity of 60 km/h. It was supposed that each UAV knew its own position and the positions of ground sensor nodes correctly.

Figure 2 illustrated the sensor nodes placement and movement during one simulation. There were 16 randomly placed nodes, the cyan square represented the first class sensor nodes which had an arrival rate of 32 kbps and the purple red diamond represented the second class sensor nodes with an arrival rate of 128 kbps data. The black triangle was used to denote the start point where the place all UAVs began their service. In order to illustrate the robustness of our algorithm towards tasks movement, we assumed there were two nodes moving after the UAVs began to collect data. Such kind of cases were suitable for collecting the data from the sensors carried by livestock[20-21].

When there was just one UAV, the path was planned according to travelling salesman problem (TSP) algorithm[22]. Figure 3 showed the flight route of single UAV.

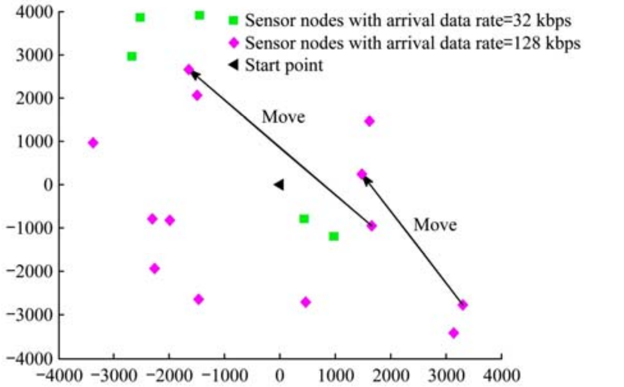

Figure 2 Sensor nodes placement and movement

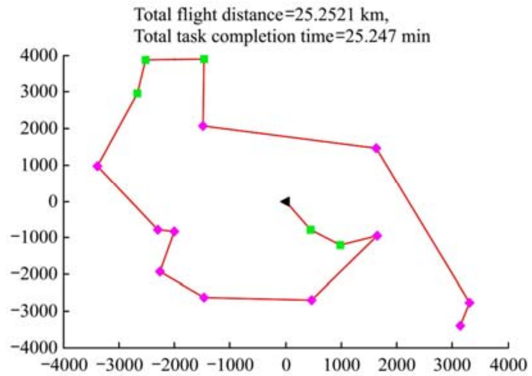

Figure 3 Single UAV's service route

Figure 4 and Figure 5 showed the tasks assignment and flight routes generated by the proposed distributed auction algorithm when there were two UAVs available. Figure 4 was for static sensor nodes and Figure 5 showed the changed routes when two sensor nodes moved. It can be seen that the UAVs could adjust their routes and even switch their tasks according to the new locations of the nodes. Note that the total task completion time represented the time that all UAVs finishing their tasks. As we considered they began their service at the same time, the total task completion time was actually the maximal one among all UAVs' task completion time. Meanwhile, the total flight distance represented the summation of the travel distance of all UAVs.

Figure 6 and Figure 7 showed the tasks assignment and the path routes for three UAVs. By comparing these two figures, we noted that the moving tasks may result in the big change to UAVs' routes and task allocation, especially when the movements happened at the beginning of the flight.

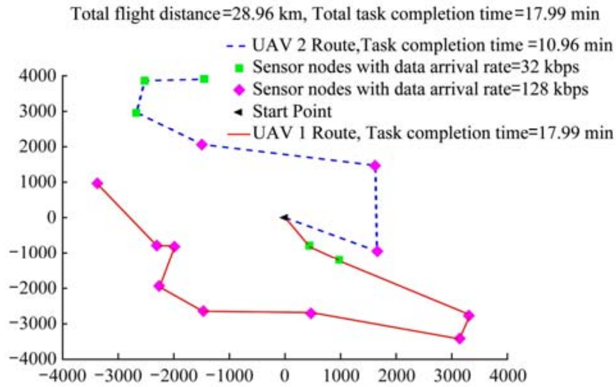

Figure 4 Two UAVs' service routes

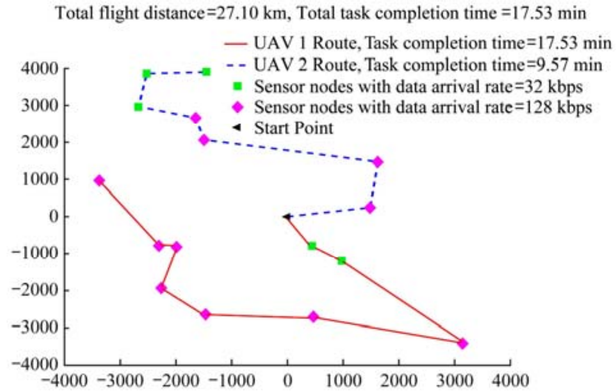

Figure 5 Two UAVs' service routes (with two moving nodes)

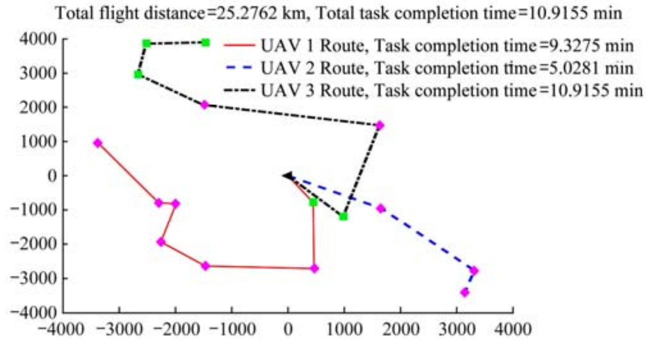

Figure 6 Three UAVs' service routes

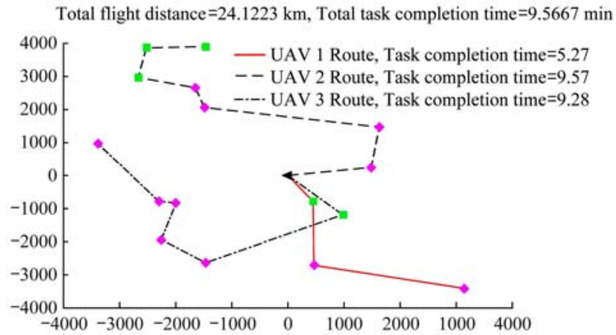

Figure 7 Three UAVs' service routes (with two moving nodes)

In Figure 3 to Figure 7, when the sensor nodes with larger data rate are not too far from the start point, UAVs would put those nodes in the priority queue. Through the comparison between Figure 3, Figure 4 and Figure 6, we could see the time efficiency of multi-UAV in performing multiple tasks. The advantage of multi-UAV to collect data could be clearly illustrated in Figure 8. The data overflow due to delay was the worst if just one UAV was available, and three UAVs could greatly help decrease the overflow.

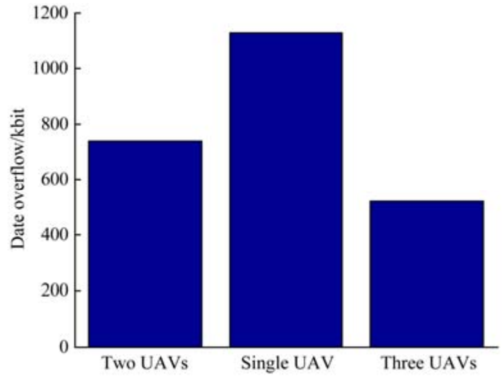

Figure 8 Sensor nodes data overflow

Next, Monte-Carlo simulations were carried to assess the performance of the proposed distributed auction algorithm in terms of average travel distance, average task completion time, average data overflow and average convergence time, as the number of tasks increased. The performance was the average of 500 simulation times.

As the number of sensor nodes were increasing, the average travel distance, the average task completion time and the average data overflow all increased. For a given number of sensor nodes, the average travel distances were almost the same for one, two UAVs or three UAVs. Whereas, more UAVs would save time and help decrease the data overflow.

In Figure 12, the convergence time was assessed for the proposed distributed auction algorithm and the centralized algorithm CSGA. Since the CSGA algorithm assigned each task one at a time, the convergence time steps for CSGA was same as the number of sensor nodes. For distributed auction, each UAV carried a bundle of tasks and bid for them, then multiple tasks could be assigned to a UAV and multiple conflicts could be resolved in parallel, thus achieved acceleration in its convergence time.

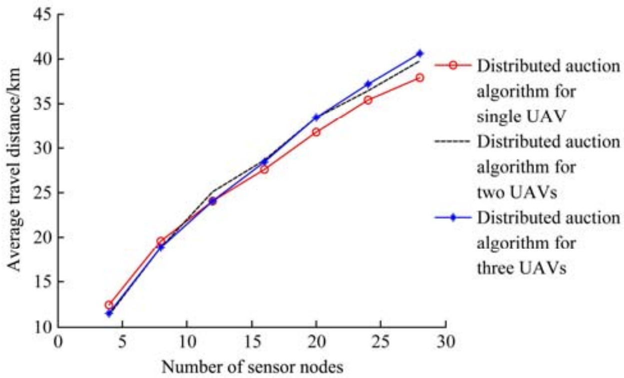

Figure 9 Total distance traveled to accomplish tasks

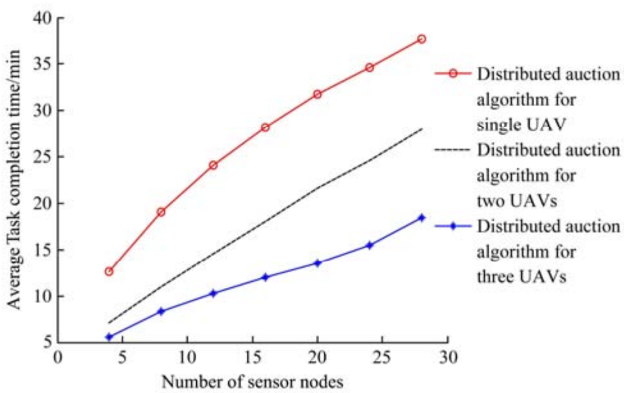

Figure 10 Average task completion time

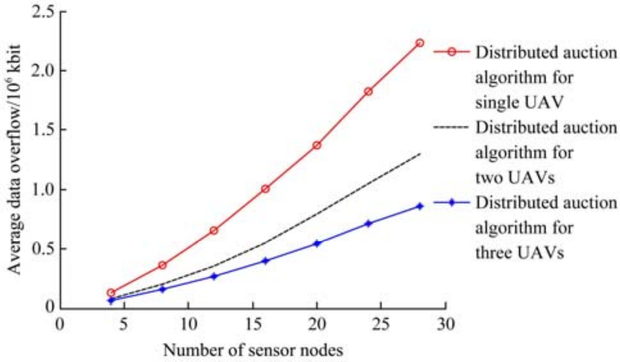

Figure 11 Average data overflow due to time delay

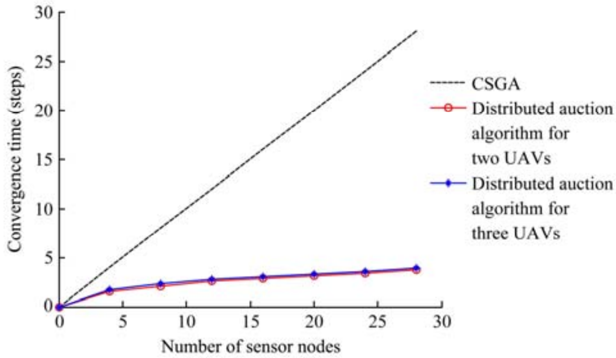

Figure 12 Convergence time steps

7 Conclusions

A distributed auction algorithm was proposed for multiple farming task allocation and route planning among multiple UAVs. The algorithm is capable of taking into account the characteristics of different farming tasks (such as moving tasks, tasks with different priority, etc.), and is capable of taking into account the limitations of different agricultural UAVs (such as flight endurance, weight load, battery capacity, data storage capacity, etc.).

By grouping similar tasks in the bundle construction stage, our algorithm converged faster than its sequential counterparts and had good performance in the assignment since they can logically group tasks that have commonalities. Furthermore, the algorithm was able to outbid earlier allocated tasks in the conflict resolution stage, which helps provide better assignments.

Moreover, the proposed algorithm simultaneously assigned the tasks and planned the flight route. By considering both problems together, our algorithm could achieve better performance than other allocate-then-plan schemes. The performance of our scheme was proved to be at least 50% optimality.

However, there are some specific constraints that haven't been taken into account in the algorithm. For example, not all regions are suitable for takeoff or landing with aerial robots. The service trajectory must ensure starting and ending points in places that fulfill some requirements, such as safety margins, sufficient space for operation and accessibility. Other constraints include forbidden zones, turn angles and sensors. Future work will focus on solving those problems and developing an integrated tool to conduct field experiments.

Acknowledgements

Project supported by the Natural Science Foundation of Guangdong Province, China (2015A030310334), Science and Technology Planning Project of Guangdong Province, China (2016A020210081) and the 111 Project, China (grant number D18019).

[References]

[1] Barrientos A, Colorado J, del Cerro J. Aerial remote sensing in agriculture: a practical approach to area coverage and path planning for fleets of mini aerial robots. Journal of Field Robotics, 2011; 28(5): 667–689.

[2] Moon S W, Shim D H C. Study on path planning algorithms for unmanned agricultural helicopters in complex environment. International Journal of Aeronautical and Space Sciences, 2009; 10(2): 1–11.

[3] Bruno S F, Fausto G C, Gustavo P, et al. The use of unmanned aerial vehicles and wireless sensor networks for spraying pesticides. Journal of Systems Architecture, 2014; 60(4): 393–404.

[4] Xue X Y, Lan Y B, Sun Z. Develop an unmanned aerial vehicle based automatic aerial spraying system. Computers and Electronics in Agriculture, 2016; 128: 58–66.

[5] Goktogan A H, Sukkarieh S, Bryson M. A rotary-wing unmanned air vehicle for aquatic weed surveillance and management. Journal of Intelligent & Robotic Systems, 2010, 57(1-4): 467–484.

[6] Polo J, Hornero G, Duijneveld, C, et al. Design of a low-cost wireless sensor network with UAV mobile node for agricultural applications. Computers and Electronics in Agriculture, 2015; 119: 19–32.

[7] Jawhar I, Mohamed N, Al-Jaroodi J. A framework for using unmanned aerial vehicles for data collection in linear wireless sensor networks. Journal of Intelligent & Robotic Systems, 2015; 74(1-2): 437–453.

[8] Sayyed A , de Araujo G M, Bodanese J P, et al. Dual-stack single-radio communication architecture for UAV acting as a mobile node to collect data in WSNs. Sensors, 2015; 15(9): 23376–23401.

[9] Huang, Y B, Thomson, S J, Hoffmann, W C , Lan, Y B. Development and prospect of unmanned aerial vehicle technologies for agricultural production management. International Journal of Agricultural and Biological Engineering, 2013; 6(3): 1–10

[10] Herwitz S R, Johnson L F, Higgins R G, et al. Imaging from an unmanned aerial: agricultural surveillance and decision support. Computers and Electronics in Agriculture, 2004; 44(1): 49–61

[11] Kwasnica A. M, Ledyard J. O, Porter D. A new and improved design for multi-object iterative auctions. Manage. Sci., 2005; 51(3): 419–434

[12] Sariel S, Balch T. Real time auction based allocation of tasks for multi-robot exploration problem in dynamic environments. In Proc. AAAI Workshop: Integrating Planning Into Scheduling, 2005; 27–33

[13] Sujit P. B, Beard R. Distributed sequential auctions for multiple UAV task allocation. inProc. 2007 American Control Conf., 2007; 3955–3960.

[14] Mercker T，Casbeer D W，Millet P T，Akella M R.An extension of consensus-based auction algorithms for decentralized, time-constrained task assignment. inProc. 2010 American Control Conference, 2010; 6324–6329

[15] Choi H L，Brunet L，How J P. Consensus-based decentralized auctions for robust task allocation. IEEE Transactions on Robotics, 2009; 25(4): 912–926.

[16] Ng H K ,Sridhar B. Computational approaches to simulation and optimization of global aircraft trajectories. Journal of Aerospace Information Systems, 2016; 13(2): 1–13.

[17] Jaggi N. Node activation policies for energy-efficient coverage in rechargeable sensor systems. Ph.D. dissertation, 2007; Rensselaer Polytechnic Institute

[18] Tang S J, Li X Y, Shen X F, et al. Cool: on coverage with solar–powered sensors. inProc. 31st International Conference on Distributed Computing Systems, 2011; 488–496

[19] Saad W, Zhu H, Basar T, et al. Hedonic coalition formation for distributed task allocation among wireless agents. IEEE Transactions on Mobile Computing, 2010; 10(9): 1327–1344.

[20] Gonzalez L A , Bishop-Hurley G, Henry D,et al. Wireless sensor networks to study, monitor and manage cattle in grazing systems. Animal Production Science, 2014; 54(10): 1687–1693

[21] Greenwood P L, Valencia, P. New ways of measuring intake, efficiency and behaviour of grazing livestock. Animal Production Science, 2014; 54(10): 1796–1804

[22] Oberlin P, Rathinam S ,Darbha S .Today's traveling salesmen problem: heterogeneous, multiple depot, multiple UAV routing problem. IEEE Robotics & Automation Magazine, 2010; 17(4): 70–77