

Chapter 69

An Improved Market-Based Auction Algorithm for UAVs Task Assignment Problem

Xin Liu and Ronggang Zhu

Abstract This paper analyzes distributed task assignment for unmanned aerial vehicles (UAVs) using a market-based parallel auction algorithm. This approach includes five steps: choosing tasks into subset, auctioning the task subset, evaluating the bids, assigning tasks to unmanned aerial vehicles and arranging the velocity of each UAV and tasks execution times. Through auctioning several tasks, including coupling and individual tasks, in one time, this method greatly reduces the communication requirements between unmanned aerial vehicles in the process of distributed task assignment, and therefore results in lower risk of being detected and higher real time property. Simulation results show that market-based parallel auction algorithm can efficiently allocate tasks to UAVs.

Keywords Distributed task assignment · UAVs · Market-based parallel auction algorithm

69.1 Introduction

With the development of artificial intelligence (AI) and the application of AI on unmanned aerial vehicles (UAVs), the capability of UAVs' autonomous determination is promoted gradually [1]. In the future, UAV teams may plan and

X. Liu (✉) · R. Zhu (✉)
Science and Technology on Electro-optic Control Laboratory,
25th West Kaixuan Road, Luoyang 471009, China
e-mail: 511065048@qq.com

R. Zhu
e-mail: ZHURG613@sina.com

execute missions without human feedback. Unmanned Systems Integrated Roadmap FY2011-1036 promulgated by the Department of Defense of the United States hopes to make UAV teams has the ability of autonomous collaborative combat before 2017.

Collaborative task assignment is the key technology of UAVs collaboration system [2, 3]. The purpose of collaborative task assignment is to assign a set of cooperative tasks to UAV team and arrange mission sequences and execution times for every UAV in the team. Distributed task assignment based on multi-agent technology can utilize every team member's computation capability and make the system robust and flexible [4]. To achieve the goal of autonomous collaboration, we require the agents to be able to compute a coordinated task strategy and share the available data.

Market-based approach is suitable for distributed task assignment [5]. One of the representative market-based approaches is auction algorithm [6]. Auction algorithm assigns task to UAV which has the greatest superiority through processes of tender and bidding. But the traditional one-by-one auction algorithm has one disadvantage that is only one task will be assigned in one auction process. So the communication demands among UAVs are very huge. For the existence of communication delays, frequent communication will affect the real time property of this approach. At the same time, frequent communication increases the risk of be detected by enemy.

This paper presents an improved market-based auction algorithm to solve the problem of allocating tasks including coupled ones to multiple UAVs. The outline of this paper is as follows. After the introduction, the scenario of collaborative task assignment for UAV team is showed (Sect. 69.2). Section 69.3 describes the improved market-based auction algorithm in detail. After the simulation results, presented in Sects. 69.4 and 69.5 reports the conclusions and future developments.

69.2 Scenarios

Given a list of N_T tasks and a UAV team formed by N_U UAVs, the goal of the task assignment is to find a conflict-free matching of tasks to UAVs that maximizes performance of the whole system. An assignment is said to be conflict-free if each task is assigned to no more than one UAV [7, 8]. Some tasks have close collaboration requirements that constraint tasks to be carried out simultaneously. Every task has an earliest execution time point ET_j and a latest execution time point LT_j which limit the task to be carried out in the time bounds $[ET_j, LT_j]$. There are N_D dangerous areas defended by ground-to-air missile or antiaircraft artillery in the planning environment across which UAVs are forbidden to fly. The performance of the whole system is assumed to be a sum of local reward values, while each local reward is determined as a function of tasks assigned to each UAV.

The task assignment problem described above can be written as the following program with binary decision variables x_{ij} that indicate whether or not task j is assigned to UAV i :

$$\text{Min} \quad \sum_{i=1}^{N_U} \sum_{j=1}^{N_T} cost_{ij} x_{ij} \quad (69.1)$$

$$\text{Subject to :} \quad \sum_{j=1}^{N_T} x_{ij} \leq M_{iT} \quad (69.2)$$

$$\sum_{i=1}^{N_U} x_{ij} \leq 1 \quad (69.3)$$

$$t_j \in [ET_j, LT_j] \quad (69.4)$$

$$x_{ij} \in \{0, 1\} \quad (69.5)$$

where $x_{ij} = 1$ if task j is assigned to UAV i and c_{ij} is a nonnegative function of assignment x_{ij} . $cost_{ij}$ is the cost of UAV i to carry out task j includes the length of route UAV i flying to task j location and the risk of being shot down if it cross enemy antiaircraft weapon defense areas. If UAVs are enforced to avoid the threat areas then $cost_{ij}$ is just the length cost. t_j is the estimate time when task j is carried out, which should be in the time window $[ET_j, LT_j]$.

It is assumed in this paper that UAVs in the team are isomorphic, that is when not think about the consumption of fuel each UAV has the same probability of finishing the same task successfully [9].

69.3 Algorithms

Market-based auction algorithm is suitable for distributed multi-agent task assignment scene for it does not require each agent in the team know about the global information. But the auction algorithm is inefficient when the size of task assignment is large because of abundant deals and communication between agents [10]. Another disadvantage of auction algorithm is that coupling constraints between tasks are not taken into account when tasks are allocated.

To satisfy the need of efficient cooperative task assignment, a parallel auction (PA) algorithm is presented. PA algorithm auctions a subset of tasks includes coupling and separated tasks in one time. The decrease of auction times reduces the communication requirements between agents. And it is easy to deal with collaborative demands when coupling tasks are auctioned together. The details of PA algorithm are as follows.

69.3.1 Subset of Tasks

Tasks are picked up and compose a subset of tasks waiting for auction before executing the PA algorithm. Every task which has not be allocated is endued with a priority defined as

$$pri_T_j = 1 - \frac{LT_j}{\max_{i=1}^{N_T} LT_j} \quad (69.6)$$

Equation (69.6) reflects the urgency of tasks. The earlier of the latest start time point, the higher of the priority.

The agent who is in charge of the auction picks up the task with highest priority T_k into subset. Then the close coupling tasks of the highest priority task T_k are sent to subset. The number of tasks in the subset N_{ST} should not larger than N_{CU} . N_{CU} is the number of usable UAVs in the team. If $N_{ST} < N_{CU}$, no more than $N_{CU} - N_{ST}$ tasks which have lower priority will be chosen into the subset.

All the tasks in the subset will be auctioned together. Each agent can get no more than one task in one auction process.

69.3.2 Auction

After the subset of tasks is collected, the principal agent sends task information of the subset to all the agents in the team. Team members respectively work out practical scheme for every task in the subset. A scheme bid_{ij} consists of three parts: cap_{ij} showing the capability of agent i to execute task j , $length_{ij}$ computing the length of route through which agent i flies to the position of task j and $[\min_rt_{ij}, \max_rt_{ij}]$ estimating time bound of agent i to execute task j .

$$bid_{ij} = \{cap_{ij}, length_{ij}, [\min_rt_{ij}, \max_rt_{ij}]\} \quad (69.7)$$

$bid_i = \{bid_{ij} | j = 1, 2, \dots, N_{ST}\}$ is the aggregate of agent i 's scheme.

To calculate bid_{ij} , the route planning system of the agent i plans a feasible route $route_{ij}$ from the position of the latest task in its task schedule to task j 's position and route $route_{jo}$ from the position of task j to the reclaiming airport. We can get $length_{ij}$ and $[\min_rt_{ij}, \max_rt_{ij}]$ via $route_{ij}$ and the velocity range of agent i and get $length_{jo}$ via $route_{jo}$. The capability of agent i fulfilling task j can be defined as

$$cap_{ij} = \begin{cases} 0, & length_{ij} + length_{jo} > rest_length_i \\ 0, & weapon_i > rest_weapon_i \\ v_length_{ij} \times (1 - \beta) + v_weapon_{ij} \times \beta, & others \end{cases} \quad (69.8)$$

$$v_length_{ij} = 1 - \frac{length_{ij} + length_{jo}}{length_i} \quad (69.9)$$

$$v_weapon_{ij} = 1 - \frac{weapon_j}{rest_weapon_i} \quad (69.10)$$

where $rest_length_i$ is the length that agent i can fly by its rest fuel and $rest_weapon_i$ is the qualities of rest weapons after agent i finishes its task schedule. The more fuel and weapons remain means agents have the higher possibility to execute one task again if the first execution activity is failed.

69.3.3 Evaluate

Every member of the team sends back its bid to the principal agent. The principal agent evaluates all the bids together. The results constitute a $N_{CU} \times N_{ST}$ superiority matrix Sup where Sup_{ij} reflects the superiority of agent i competing with other agents for task j . The measurement equation of Sup_{ij} is

$$Sup_{ij} = \begin{cases} 0, & cap_{ij} = 0 \\ [cap_{ij} \times (1 - \rho) + pro_len_{ij} \times \rho] \times ctime_{ij}, & cap_{ij} > 0 \end{cases} \quad (69.11)$$

where

$$pro_len_{ij} = 1 - \frac{length_{ij}}{\max_{i=1}^{N_V}(length_{ij})} \quad (69.12)$$

and

$$ctime_{ij} = \begin{cases} 0, & \min_rt_{ij} > LT_j \\ 0.5, & \max_rt_{ij} < ET_j \\ 1, & [\min_rt_{ij}, \max_rt_{ij}] \cap [ET_j, LT_j] \neq \emptyset \end{cases} \quad (69.13)$$

pro_len_{ij} is the route length factor that makes the agent i near the position of task j has a high probability of being chosen to execute task j . But it does not mean that the nearest agent has the highest probability to execute the task. For every task has special executing time window $[ET_j, LT_j]$, $\min_rt_{ij} > LT_j$ means even the agent i flies to task position with the highest velocity it cannot arrive the task location before the deadline of the task j and $\max_rt_{ij} < ET_j$ is the case that even the agent i flies to task position with the lowest velocity it will get to the task location before ET_j . $ctime_{ij}$ is a time punishment factor. In the first case above, agent i can not finish task j within the prescriptive time bounds, so $ctime_{ij}$ is set as 0. In the second case agent i has to plan a waiting route and cruise until fit the time requirement. To punish the waste of fuel the time punishment factor is set as 0.5. When $[\min_rt_{ij}, \max_rt_{ij}]$ and

$[ET_j, LT_j]$ have intersection, which means agent i will get to task position between $[ET_j, LT_j]$ by adjusting its flying velocity, $ctime_{ij}$ will be set as 1.

The length of route, time fitness and the capability of agent i performing task j have been taken into account in the computation of Sup_{ij} . Sup_{ij} is a composite result of three factors mentioned above. β is an accommodation coefficient with a range from 0 to 1 which reflects decision-maker's favor. For all the factors in the function have a bound of $[0, 1]$, the magnitude of Sup_{ij} is between 0 and 1.

69.3.4 Assignment

As mentioned in Sect. 69.3.1, each agent only can get no more than one task in one auction and each task can be distributed to one agent. The problem of assigning UAVs to tasks in the subset is defined as

$$\text{Max} \quad \sum_{i=1}^{N_{CU}} \sum_{j=1}^{N_{ST}} Sup_{ij} x_{ij} \quad (69.14)$$

$$\text{Subject to :} \quad \sum_{j=1}^{N_{ST}} x_{ij} \leq 1 \quad (69.15)$$

$$\sum_{i=1}^{N_{CU}} x_{ij} \leq 1 \quad (69.16)$$

$$x_{ij} \in \{0, 1\} \quad (69.17)$$

Hungary algorithm has been designed to solve such problem [11]. But the normal Hungary algorithm can deal with the case that N_{ST} equals to N_{CU} . If $N_{CU} > N_{ST}$, we can set $N_{CU} - N_{ST}$ fictitious tasks and consider superiority of every UAV bidding those fictitious tasks as 0.

$$Sup_{ij} = 0, \quad i = 1, 2, \dots, N_{CU}; j = N_{ST} + 1, \dots, N_{CU} \quad (69.18)$$

After those fictitious tasks join the subset, the dimension of superiority matrix is $N_{CU} \times N_{CU}$. Then we can get a $N_{CU} \times N_{CU}$ 0-1 matrix R shows the allocation result computed by normal Hungary algorithm. Delete the last $m - n$ lists of R and we will get the final assigning result.

69.3.5 Velocity and Execution Time Arrangement

The above steps insure that tasks can be carried out in special time windows. Then we need to fix the practical execution time point of each task and the flying velocity of each UAV. In this part, the data got from Sect. 3.3 will be useful.

Assume that agent i is assigned to individual task j , we will have two segments of time $[\min_rt_{ij}, \max_rt_{ij}]$ and $[ET_j, LT_j]$. Through evaluation in part C and assignment in part D the situation of $\min_rt_{ij} > LT_j$ is avoided. If $\max_rt_{ij} < ET_j$, ET_j is the time point for agent i to execute task j , and agent i should fly to task position with its lowest velocity V_{\min} . In this case a path of waiting route is needed and the length of waiting route is calculated by Eq. (69.19).

$$wlength_{ij} = V_{\min} \times (ET_j - \min_rt_{ij}) \quad (69.19)$$

If $[\min_rt_{ij}, \max_rt_{ij}] \cap [ET_j, LT_j] \neq \emptyset$ and the intersection is $[at_j^{\min}, at_j^{\max}]$, at_j^{\min} will be the execution time of task j and flying velocity of agent i in this section is defined as

$$v_{ij} = \frac{length_{ij}}{at_j^{\min} - pretime_i} \quad (69.20)$$

where $pretime_i$ is the estimate time that agent i finish its task sequence.

It is analogous to get the execution times of simultaneous tasks. Generally, simultaneous tasks have the same time windows. Assume that agent k and agent l are assigned to carrying out simultaneous tasks with time window $[ET_m, LT_m]$, $[\min_rt_k, \max_rt_k]$ and $[\min_rt_l, \max_rt_l]$ are the estimating time bounds of agent k and agent l to carry out those simultaneous tasks mentioned above. For the usage of waiting route and waiting time, the latest time that agent k and agent l carry out collaborative tasks can be set to LT_m . ct is defined as $ct = \max\{\min_rt_k, \min_rt_l\}$. If $ct \in [ET_m, LT_m]$, the execution time equals to ct , else it equals to ET_m . The computation of waiting length and flying velocity is the same as Eqs. (69.19) and (69.20).

69.4 Simulation Examples

To clarify the scenarios and illustrate the assignment algorithm presented in Sect. 69.3, a simulation example with a problem of size $N_U = 3$ and $N_T = 8$ is presented here. UAVs and task positions are placed in an area 100 km wide and 100 km long. Four detected and some undetected antiaircraft threats with different threaten ranges exist in the planning environment. The parameters of UAVs, tasks and threats are showed in Tables 69.1, 69.2 and 69.3 respectively. Time windows are set to the bounds from the start time point of the assignment process.

Table 69.1 The parameters of UAVs

UAV	Original location (X, Y)/km	Velocity/km/h	Weapon units	Rest length/km
1	(5.00, 15.00)	[80, 150]	3	300
2	(15.00, 5.00)	[80, 150]	3	300
3	(5.00, 5.00)	[80, 150]	4	300

Table 69.2 The parameters of tasks

Task	Location (X, Y)/km	Time window/min	Simultaneous task
1	(50.12, 42.22)	[20, 30]	
2	(32.62, 60.89)	[30, 40]	2
3	(32.62, 60.89)	[30, 40]	3
4	(60.76, 73.77)	[55, 65]	
5	(80.11, 50.99)	[45, 55]	
6	(85.65, 40.76)	[40, 50]	
7	(90.11, 92.90)	[70, 80]	8
8	(90.11, 92.90)	[70,80]	7

Table 69.3 The parameters of threats

Threat	Location (X, Y)/km	Range/km
1	(24, 45)	12
2	(45, 25)	15
3	(64, 50)	12
4	(80, 74)	15
5	(50, 80)	12

We assume that communications between UAVs are perfect, without delay and information losses. One of the team members is chosen as the tasks owner and host auctions. Each UAV has a route planner planning the best flying routes between two positions with sparse A* (SAS) algorithm [12] and task time availability windows are computed based on computing the length of the best flying routes. The progression of the simulation is shown in Fig. 69.1.

Figure 69.1a shows the UAV and task positions at the beginning of the simulation. UAV team is cruising in the battle field. Figure 69.1b shows the first round auction result. Task1, 2 and 3 are chosen into the subset of tasks. After the processes of auction, evaluation and assignment presented in Sect. 69.3, Task1, 2 and 3 are assigned to UAV3, 1 and 2 respectively. The colorized lines are the best flying routes planned by SAS algorithm. Figure 69.1c, d show the second and third round auction result.

The velocity of each UAV in different phases and estimating execution time of each task is computed by method presented in Sect. 69.3.5. The arrangement results are shown in Table 69.4.

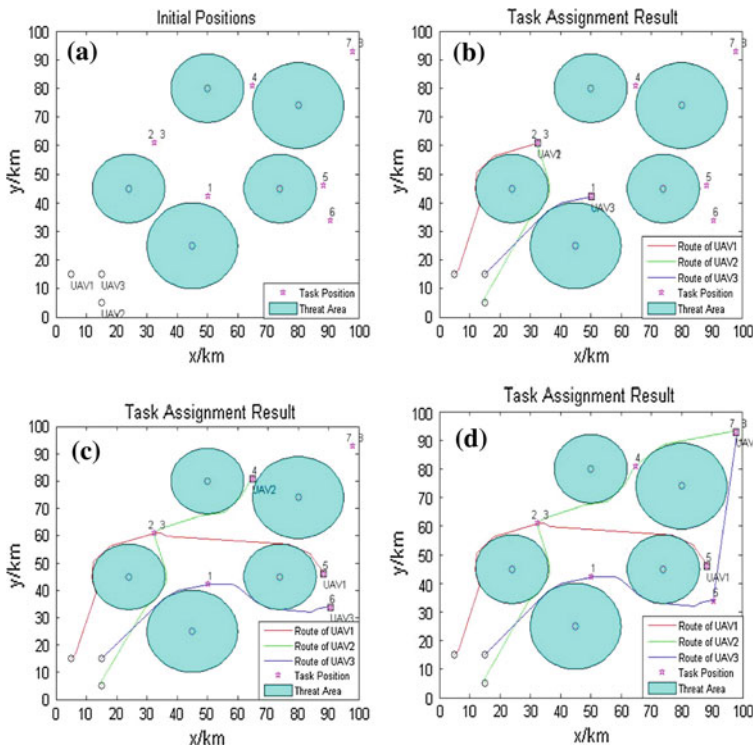


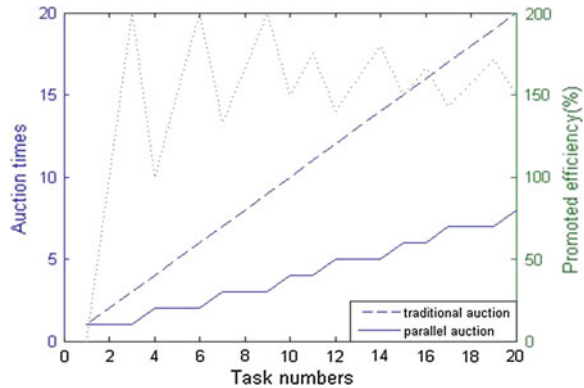
Fig. 69.1 Simulation example. **a** Initial positions. **b** First round auction results. **c** Second round auction results. **d** Third round auction results

Table 69.4 The arrangement of execution times and velocity

UAV	Task	Execution time/min	Velocity/km/h	Length of route/km
1	2	30.0000	118.4850	59.2425
1	5	54.2236	150.0000	60.5589
2	3	30.0000	123.0209	61.5105
2	4	55.0000	92.2935	38.4556
2	8	73.6703	113.7336	35.3906
3	1	20.0000	135.9763	45.3254
3	6	40.0000	130.2465	43.4155
3	7	73.6703	104.3601	58.5639

From the simulation results we can find eight tasks are assigned to three UAVs by three times of auction activities, less than eight times of auction activities of the traditional one-by-one auction algorithm. So the efficiency is promoted nearly 167 %. To calculate the efficiency promoted by parallel auction algorithm, a series of tests was carried out in the same battle scene with 1–20 tasks. 3 UAVs with sufficient fuel and weapons join the tests. Respectively run the traditional auction

Fig. 69.2 Parallel auction algorithm compared with traditional auction algorithm in auction times



algorithm and the improved parallel auction algorithm and record the auction times. The results are showed in Fig. 69.2. We can see that the efficiency is promoted 150–200 %.

69.5 Summaries

The problem of assigning collaborative and individual mixed tasks to multiple UAVs has been presented. The problem was solved using an improved market-based PA algorithm. This methodology allows the host agent to auction a series of tasks in one time therefore greatly decreasing the number of deals compared to traditional market-based auction algorithm. The reduction in number of auctions results in decrease in communication cost. The simulation results show that PA algorithm presented is realizable and has higher efficiency than traditional auction algorithm.

References

1. Joint Planning and Development Office. NextGen UAS research, development and demonstration roadmap (2012)
2. Shima T, Rasmussen S (2008) UAV cooperative decision and control. Society of Industrial and Applied Mathematics, Philadelphia
3. Liao M, Chen Z (2007) Dynamic target assignment method based on multi-agent decentralized cooperative auction. *J Beijing Univ Aeronaut Astronaut* 33(2):180–183 w(in Chinese)
4. Long T (2006) Research on distributed task allocation and coordination for multiple UCAVs cooperative mission control. Graduate School of National University of Defense Technology, Changsha (in Chinese)
5. Gao P (2009) Market-based approach to multi-robot task allocation. Central South University, Changsha (in Chinese)

6. Sujit PB, Beard R (2007) Multiple MAV task allocation using distributed auctions. In: AIAA guidance, navigation and control conference and exhibit, Hilton Head. AIAA 2007-6452
7. Yang L, Kreamer W, Adams M et al (2004) Hierarchical planning for large numbers of unmanned vehicles. In: AIAA 1st intelligent systems technical conference, Chicago. AIAA 2004-6246 (in Chinese)
8. Vandermeersch BRR, Chu QP, Mulder JA (2005) Design and implementation of a mission planner for multiple UCAVs in a SEAD mission. In: AIAA guidance, navigation, and control conference and exhibit, San Francisco. AIAA 2005-6480
9. Shaferman V, Shima T (2009) Task assignment and motion planning for multiple UAVs tracking multiple targets in urban environments. In: AIAA guidance, navigation, and control conference, Chicago. AIAA 2009-5778
10. Li Y (2008) Research on coalition and task allocation in multi-agent system. Hefei University of Technology, Hefei (in Chinese)
11. Xu Y, Zhang H (2006) Operational research. Peking University Press, Beijing (in Chinese)
12. Ding M, Zheng C, Zhou C et al (2009) Route planning for unmanned aerial vehicles. Publishing House of Electronics Industry, Beijing (in Chinese)