# Autonomous Mission Management for Forest Search with Multiple Unmanned Aerial Vehicles

Kemao Peng[1], Feng Lin[1] and Ben M. Chen[2]
[1]Temasek Laboratories, National University of Singapore, 117411, Singapore, Singapore
[2]Department of Electrical and Computer Engineering, National University of Singapore, 117576, Singapore, Singapore

Keywords: Unmanned Aerial Systems, Autonomy of UAVs, Multiple UAVs.

Abstract: An autonomous mission management (AMM) system is designed with the enhanced hierarchical-distributed methodology (HDM) for multiple unmanned aerial vehicles (UAVs) to search a field of forest together. The main ideas of the enhanced HDM are hierarchical control and distributed implementation. The event control law is partitioned into the group and individual event control laws. The group event control law is to coordinate the group of UAVs to complete the designated mission and the individual event control laws are to complete the assigned submissions/ tasks accordingly. The group event control law is executed by the leader and any member can be designated or selected as the leader on the rules. The forest search is applied to verify the designed AMM system in simulation. The simulation results demonstrate that the designed AMM system is successful to complete the designated mission by collaborating the group of UAVs.

## 1 INTRODUCTION

The autonomous mission management (AMM) has been attracting much attention for the multiple unmanned aerial vehicles (UAVs) as it is necessary to coordinate a group of UAVs to complete a designated mission together (Bellingham et al., 2002), (Inalhan et al., 2002), (Peng et al., 2014), (Richards et al., 2002). Such the AMM processes in the high level and the dynamic processes in the low level of the multiple UAVs construct a typical hybrid system in which the dynamic processes are time-driven and the AMM processes are event-driven (Kaminer et al., 2006), (Tomlin et al., 2000), (Uhrmann and Schulte, 2011), (van der Schaft and Schumacher, 1998). The event variables are defined to describe the system behaviors that are extracted from the time-driven process. The transition of the discrete event states will become part of the event control laws to be designed. Thus, too many defined event variables may result in that the designed event control laws are not implementable.

Many researchers dedicated their effort to the hybrid systems (Jadbabaic et al., 2003), (Kopeikin et al., 2013), (Lafferriere et al., 2005), (Murata, 1989), (Teo et al., 2004), (Wong-Toi, 1997), (Ye et al., 1998). The typical tools such as Petri net and temporal logic are popularly used in the logic control design of the hybrid systems. However, the core problems are still left to study. One problem is how to describe the system behaviors with reasonable number of the event variables so that the designed event control laws are implementable. Another problem is the conversion of the event commands to dynamic commands as the UAVs are capable only of tracking the dynamic commands. Such conversion is subject to the capabilities of the UAVs. Otherwise, the event commands are not executable.

Our objective is to design an AMM system to collaborate a group of UAVs (quadrotors) to search a field of forest together. Such system is described in the hybrid systems. We try to attenuate the effect of the two problems by reducing number of the event variables to be defined and defining available event commands based on the physical process. This motivated us to propose the enhanced hierarchical - distributed methodology (HDM).

HDM was proposed for formation flight in (Peng et al., 2014) . In this paper, it is enhanced for more complex scenarios so-called the enhanced HDM in which a submission/ task is decomposed into a series and/or parallel of tasks/ subtasks respectively as the UAVs can execute various kinds of tasks/ subtasks simultaneously. Such enhancement results in that the missions can be decomposed perfectly with the series and/ or parallel relationships in multiple levels. The main ideas of the enhanced HDM are hierarchi-

Peng, K., Lin, F. and Chen, B.
Autonomous Mission Management for Forest Search with Multiple Unmanned Aerial Vehicles.
DOI: 10.5220/0005949603850392
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 385-392
ISBN: 978-989-758-198-4

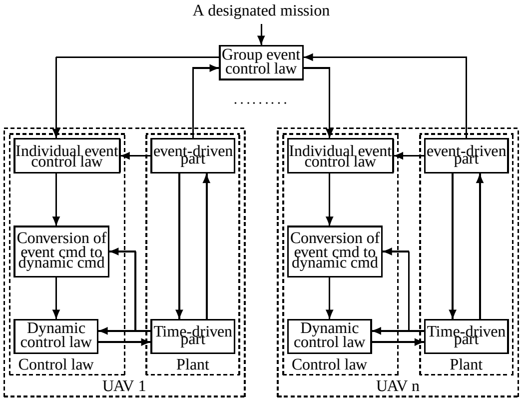


Figure 1: A closed-loop system of multiple UAVs.

cal control and distributed implementation. The event control law is divided into the group and individual event control laws. The group event control law is to focus on the group events to collaborate the group of UAVs to complete the designated mission together. The individual event control laws are to focus on the individual events to complete the assigned submissions/ tasks accordingly. The group event control law is conducted by the leader. Any group member can be designated or selected as the leader on the rules.

# 2 ENHANCED HDM

The enhanced HDM is introduced. The resulting closed-loop system is shown in Figure 1. A group of UAVs collaborate to complete a designated mission together. The event control law consists of the group and individual event control laws hierarchically. The group event control law is conducted by any member when it is designated or selected as the leader. We mainly consider the group and individual event control laws, conversion of the event commands to dynamic commands and the event-driven transition.

## 2.1 Group Event Control Design

The group event control law is focused on the group events such as the mission decomposition, schedule, assignment and progress. When an accident happens, the mission reschedule is to be considered.

### 2.1.1 Mission Decomposition

A mission may be decomposed into a series and/or parallel of submissions so that it is easy to be completed by a group of the UAVs together. The mission is completed when those submissions are completed accordingly.

$$m_{\mathrm{sn}} := \{s_{\mathrm{mn},11}, \cdots, s_{\mathrm{mn,ij}}, \cdots, s_{\mathrm{mn},m_\mathrm{s}n_\mathrm{s}}\}, \qquad (1)$$

where $m_{\mathrm{sn}}$ denotes the mission decomposition and $s_{\mathrm{mn,ij}}$ denotes the i-th and j-th submission. There are $m_\mathrm{s} \times n_\mathrm{s}$ of submissions.

### 2.1.2 Mission Schedule

A decomposed mission can be scheduled according to the series and/or parallel relationships between/ among the submissions so that the submissions can be assigned accordingly.

$$\begin{aligned} &m_{\mathrm{sch}} = \begin{pmatrix} s_{\mathrm{mn},1} & \cdots & s_{\mathrm{mn},n_\mathrm{s}} \end{pmatrix}, \\ &s_{\mathrm{mn,k}} = \begin{pmatrix} s_{\mathrm{mn,1k}} & \cdots & s_{\mathrm{mn},m_\mathrm{s}\mathrm{k}} \end{pmatrix}', \\ &k \in \{1, \cdots, n_\mathrm{s}\}, \end{aligned} \qquad (2)$$

where $m_{\mathrm{sch}}$ denotes the mission schedule. The submissions are assigned in $n_\mathrm{s}$ of steps and $m_\mathrm{s}$ of the submissions are assigned in the k-th step based on the event activation conditions.

In the presentation, the row vectors mean series and the column vectors mean parallel. A series of the submissions are completed one by one and a parallel of the submissions in a step are completed in the step. A zero element of the vectors means no corresponding assignment. No assignment and no activation are regarded as completed. The rules are applicable to presentation of the tasks and subtasks hereafter.

### 2.1.3 Mission Assignment and Progress

The mission assignment and progress can be recorded in sequence of the mission schedule so that the completed, being completed and to be completed submissions are presented clearly.

$$\begin{aligned} m_{\mathrm{ap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}, \\ m_{\mathrm{sap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}', \end{aligned} \qquad (3)$$

where $m_{\mathrm{ap}}$ denotes the step progress of the mission and $m_{\mathrm{sap}}$ denotes the progress of the submissions in the being completed step. The element 2 means that the corresponding step/ submission is completed; 1 means that the corresponding step/ submission is being completed; 0 means that the corresponding step/ submission is to be completed. The next step/ submissions to be assigned is/are clear based on the event activation conditions. The meanings of the elements are applicable to the submission/ task assignment and progress hereafter.

## 2.2 Individual Event Control Design

The individual event control laws are focused on the individual events such as the submission/ task decomposition, schedule, assignment and progress. when an accident happens, the submission/ task reschedule is to be considered.

#### 2.2.1 Submission/Task Decomposition

A submission/ task may be decomposed into a series and/or parallel of tasks/ subtasks respectively so that it is easy to be completed by a UAV. The submission/ task is completed respectively when the tasks/ subtasks are completed accordingly. The subtasks are defined as those easily completed by the UAVs.

$$\begin{aligned} s_{\mathrm{mn}} &:= \{t_{\mathrm{sk},11}, \cdots, t_{\mathrm{sk},\mathrm{ij}}, \cdots, t_{\mathrm{sk},\mathrm{m_t n_t}}\}, \\ t_{\mathrm{sk}} &:= \{s_{\mathrm{tk},11}, \cdots, s_{\mathrm{tk},\mathrm{ij}}, \cdots, s_{\mathrm{tk},\mathrm{m_{st} n_{st}}}\}, \end{aligned} \tag{4}$$

where $s_{\mathrm{mn}}$ denotes the submission decomposition and $t_{\mathrm{sk,ij}}$ denotes the i-th and j-th task. There are $m_{\mathrm{t}} \times n_{\mathrm{t}}$ of tasks. $t_{\mathrm{sk}}$ denotes the task decomposition and $s_{\mathrm{tk,ij}}$ denotes the i-th and j-th subtask. There are $m_{\mathrm{st}} \times n_{\mathrm{st}}$ of subtasks.

#### 2.2.2 Submission/Task Schedule

A decomposed submission/ task can be scheduled according to the series and/or parallel relationships between/ among the tasks/ subtasks so that the tasks/ subtasks can be assigned accordingly.

$$\begin{aligned} & s_{\mathrm{sch}} = \begin{pmatrix} t_{\mathrm{sk},1} & \cdots & t_{\mathrm{sk},\mathrm{n_t}} \end{pmatrix}, \\ & t_{\mathrm{sk,k}} = \begin{pmatrix} t_{\mathrm{sk,1k}} & \cdots & t_{\mathrm{sk},\mathrm{m_t k}} \end{pmatrix}', \\ & k \in \{1, \cdots, n_{\mathrm{t}}\}, \\ & t_{\mathrm{sch}} = \begin{pmatrix} s_{\mathrm{tk},1} & \cdots & s_{\mathrm{tk},\mathrm{n_{st}}} \end{pmatrix}, \\ & s_{\mathrm{tk,j}} = \begin{pmatrix} s_{\mathrm{tk,1j}} & \cdots & s_{\mathrm{tk},\mathrm{m_{st} j}} \end{pmatrix}', \\ & j \in \{1, \cdots, n_{\mathrm{st}}\}, \end{aligned} \tag{5}$$

where $s_{\mathrm{sch}}$ denotes the submission schedule. The tasks are assigned in $n_{\mathrm{t}}$ of steps and $m_{\mathrm{t}}$ of the tasks are assigned in the k-th step based on the event activation conditions. $t_{\mathrm{sch}}$ denotes the task schedule. The subtasks are assigned in $n_{\mathrm{st}}$ of steps and $m_{\mathrm{st}}$ of the subtasks are assigned in the j-th step based on the event activation conditions.

#### 2.2.3 Submission/Task Assignment and Progress

The submission/ task assignment and progress can be recorded in sequence of the submission/ task schedule so that the completed, being completed and to be completed tasks/ subtasks are presented clearly.

$$\begin{aligned} s_{\mathrm{ap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}, \\ s_{\mathrm{tap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}', \\ t_{\mathrm{ap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}, \\ t_{\mathrm{sap}} &= \begin{pmatrix} 2 & \cdots & 2 & 1 & 0 & \cdots & 0 \end{pmatrix}', \end{aligned} \tag{6}$$

where $s_{\mathrm{ap}}$ denotes the step progress of the submission and $s_{\mathrm{tap}}$ denotes the progress of the tasks in the being completed step. $t_{\mathrm{ap}}$ denotes the step progress of the task and $t_{\mathrm{tap}}$ denotes the progress of the subtasks in the being completed step.

### 2.3 Conversion of Event Commands to Dynamic Commands

The output of the event control laws is to assign subtasks. Such subtasks, the event commands, need to be converted properly so that they can be easily conducted by the UAVs. There are three kinds of subtasks such as (1) message subtask: It needs to define a set of knowledge for the communication between/ among the UAVs. Once a UAV receives a message, it can understand the message and know how to respond the message; (2) action subtask: It needs to define a number of actions that the UAVs can easily execute. When a UAV is assigned an action subtask, it can know what kind of action it needs to conduct; and (3) flight subtask: When a UAV is assigned a flight subtask, it should know how to complete the flight subtask. Thus, there needs an online planning to convert the event command, a flight subtask, into the dynamic command, the trackable references to the dynamic control laws. Based on such conversion, the event command can be executed by the dynamic control laws and thus the UAVs can fly on schedule.

### 2.4 Event-driven Transition

The event state transition is activated by the states of the defined events. Most of the events are extracted from the time-driven process to describe the system behaviors and thus their states depend on the time-driven process. Therefore, it needs to define the flight subtask completion conditions so that the states of the corresponding events can transit correctly.

Based on the flight subtask completion conditions, the states of the defined events are clear. Therefore, the activation conditions of the defined events each can be established according to the event states to drive the event state transition.

The four parts construct the AMM system in the high level. This completes introduction of the enhanced HDM.

## 3 DESIGN OF AMM SYSTEM

An AMM system is designed by following the ideas of the enhanced HDM to collaborate nine UAVs (quadrotors) to search a field of forest with size of 500 × 500 meters together. The design is subject to the capabilities of the UAVs and the environment situation. Next, we proceed to design the AMM system.

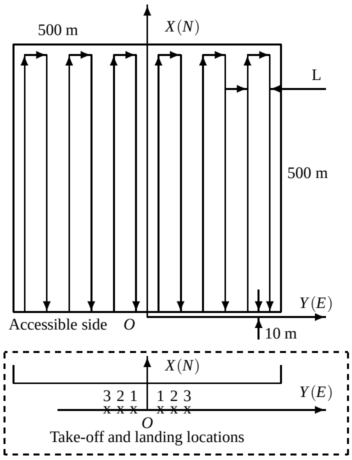


Figure 2: The forest field.

## 3.1 Group Event Control Design

The field of forest is assumed to originate normally. The North-East-Down (NED) frame is defined in Figure 2. The x denotes the take-off/ landing locations. The distance between them and the origin is 5 meters. Only one side of the forest field is accessible.

The field of forest is partitioned into a set of flight channels in Figure 2 so that the UAVs can search the forest by flying along the channels. The nominal distance between the two neighbor channels is set to $L = 40$ meters for slight overlap.

The nine UAVs are assigned in three batches and there are three UAVs in a batch. The first two batches are pre-scheduled and the third batch is online scheduled based on the search results of the first two batches. A batch of UAVs are assigned to fly into the field of forest together. Each UAV is allocated a round channel to search the forest independently and records the detected area in 2D map, the detected targets in the coordinates. All of the members in a batch meet around the launch site to hand over their data of the established 2D maps and the found targets to the batch leader before they land.

The batch leader is designated or selected on the rules. The batch leader is designated at the beginning of each batch. The first member in the first batch is designated as the leader by the ground station. The first member in the other batches is designated as the leaders by the leader of the last batch. There is no batch leader during independently searching. The first UAV that arrives at its meeting point becomes the new batch leader. A UAV is regarded lost if it cannot arrive at its meeting point in the given duration. The group resources are shared by all of members in a batch .

The batch leaders have six duties such as (1) call all of the members in the batch to take off and to search independently; (2) collect and merge those 2D maps and the found targets; (3) schedule the path online based on the merged 2D map and decide number of UAVs to be assigned in the third batch; (4) hand over the merged 2D map, found targets, online scheduled path, and assignment of the UAVs in the third batch to the standby UAVs of the next batch; (5) designate the first member of the next batch as the leader before it lands; and (6) hand over the merged 2D map and the found targets to the ground station at end of the search before it lands.

There is no communication between the UAVs and the ground station during the process except at the beginning and end of the process. At the beginning, the ground station designates the first member in the first batch as the batch leader. At the end, the ground station receives and displays the search results of the merged 2D map and the found targets. The UAVs can communicate with each other if they are close. This is the scenario of the group of UAVs.

### 3.1.1 Mission Decomposition

Based on the scenario of the group of UAVs, the mission is decomposed as follows,

$$m_{\mathrm{sn}} = \{s_{\mathrm{mn},11}, \cdots, s_{\mathrm{mn,ij}}, \cdots, s_{\mathrm{mn},33}\}, \tag{7}$$

where $s_{\mathrm{mn,ij}}$ denotes the submission to be assigned to the i-th UAV in the j-th batch. Those submissions are to be defined latter.

### 3.1.2 Mission Schedule

Based on the mission decomposition, $m_{\mathrm{sn}}$, the decomposed submissions are scheduled as follows,

$$\begin{aligned} & m_{\mathrm{sch}} = \begin{pmatrix} b_{\mathrm{atch},1} & b_{\mathrm{atch},2} & b_{\mathrm{atch},30} \end{pmatrix}, \\ & b_{\mathrm{atch},30} = \begin{pmatrix} b_{\mathrm{atch},3} & b_{\mathrm{atch},\bar{3}} \end{pmatrix}', \\ & b_{\mathrm{atch},\bar{3}} = 0, \quad \text{no assignment}, \\ & b_{\mathrm{atch,k}} = \begin{pmatrix} s_{\mathrm{mn,1k}} & s_{\mathrm{mn,2k}} & s_{\mathrm{mn,3k}} \end{pmatrix}', \\ & k \in \{1, 2, 3\}. \end{aligned} \tag{8}$$

Based on the mission schedule, $m_{\mathrm{sch}}$, nine of the submissions are assigned in three batches and three of them are assigned together in a batch based on the event activation conditions.

The mission assignment and progress vectors record the completed, being completed and to be completed batch/ submissions. The next batch/ submissions to be assigned is/are clear based on the event activation conditions. The group event state transition is shown in Figure 3. The series of the batches are are completed one by one. The third batch is activated based on the online decision. A parallel of the submissions in a batch are completed in the batch except that the assigned UAV is lost.

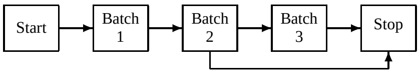


Figure 3: Group event state transition.

## 3.2 Individual Event Control Design

Each UAV is assigned a submission. The submissions are defined such as (1) receive the related data and stay on the ground; (2) take off and independently search the forest along the scheduled channel/path; (3) broadcast itself flight motion messages; (4) record the detected areas in 2D map and the found targets in the coordinates; (5) fly to and hover at the designated meeting points and attend the selection of the new batch leader; (6) hand over the recorded data to the batch leader; and (7) land on the ground and exit.

The submissions of the UAVs in the third batch are slightly revised only at the flight path that is online scheduled. The submissions of the UAVs as the batch leader are also revised. Those UAVs do not need to hand over the recorded data to the batch leader. But they have to undertake the duties of the batch leader. This is the scenario of each UAV.

### 3.2.1 Submission/ Task Decomposition

Based on the scenario of each UAV, the submissions are decomposed as follows,

$$\begin{aligned} s_{\mathrm{mn,ij}} = \{ & t_{\mathrm{sk},11}, t_{\mathrm{sk},21}, t_{\mathrm{sk},12}, t_{\mathrm{sk},22}, t_{\mathrm{sk},32}, t_{\mathrm{sk},13}, \\ & t_{\mathrm{sk},23}, t_{\mathrm{sk},4}, t_{\mathrm{sk},15}, t_{\mathrm{sk},25}\}, \;\; i,j \in \{1,2,3\}, \end{aligned} \tag{9}$$

where the decomposed tasks are listed in Table 1. Note that $s_{\mathrm{mn}}$ denotes submission, $t_{\mathrm{sk}}$ denotes task and $s_{\mathrm{tk}}$ denotes subtask.

Table 1: Decomposed tasks.

| Task No. | Tasks |
|---|---|
| 11 | Stay on the geound |
| 21 | Rv the related data |
| 12 | Take off/ search |
| 22 | Tr the motion messages |
| 32 | Record the 2D map/targets |
| 13 | Hv at the meeting point |
| 23 | Hand over the data |
| 4 | Landing |
| 15 | Landed on the ground |
| 25 | Report the landed |
| Rv | receive |
| Tr | transimit |
| Hv | hovering |

Table 2: Decomposed subtasks in Task 21.

| Subtask No, | Subtasks |
|---|---|
| 1 | Rv path/Rp |
| 2 | Rv agn/Rp |
| 3 | Rv 2D map/Rp |
| 4 | Rv tgt data/Rp |
| 5 | Rv leadership/Rp |
| 6 | Tr take-off cmd/Rv Rp |
| 7 | Tr group data |
| 8 | Rv take-off cmd/Rp |
| 9 | Rv group data |
| Rp | respond/resoinse |
| cmd | command |
| agn | assignment |
| tgt | target |

The tasks may further be decomposed as follows,

$$\begin{aligned} & t_{\mathrm{sk},21} = \{s_{\mathrm{tk1},1}, \cdots, s_{\mathrm{tk1},9}\}, \\ & t_{\mathrm{sk},12} = \{s_{\mathrm{tk2},1}, \cdots, s_{\mathrm{tk2},n_{\mathrm{st2}}}\}, \\ & t_{\mathrm{sk},23} = \{s_{\mathrm{tk3},1}, \cdots, s_{\mathrm{tk3},17}\}, \\ & t_{\mathrm{sk},4} = \{s_{\mathrm{tk4},1}, s_{\mathrm{tk4},2}\}, \end{aligned} \tag{10}$$

where $n_{\mathrm{st2}} = 7$ or to be determined online. Task 12 consists of $n_{\mathrm{st2}}$ line segments at which each segment is as a flight subtask.Task 4 consists of 2 line segments. Task 21 and Task 23 consist of the message subtasks which are listed in Table 2 and 3 respectively. The other tasks are as simple as the subtasks and thus no decomposition is needed.

### 3.2.2 Submission/ Task Schedule

Based on the submission decomposition, $s_{\mathrm{mn,ij}}$, the decomposed tasks are scheduled as follows,

$$\begin{aligned} & s_{\mathrm{sch,ij}} = \begin{pmatrix} t_{\mathrm{sk},1} & t_{\mathrm{sk},2} & t_{\mathrm{sk},3} & t_{\mathrm{sk},4} & t_{\mathrm{sk},5} \end{pmatrix}, \\ & t_{\mathrm{sk},1} = \begin{pmatrix} t_{\mathrm{sk},11} & t_{\mathrm{sk},21} \end{pmatrix}', \;\; i,j \in \{1,2,3\}, \\ & t_{\mathrm{sk},2} = \begin{pmatrix} t_{\mathrm{sk},12} & t_{\mathrm{sk},22} & t_{\mathrm{sk},23} \end{pmatrix}', \\ & t_{\mathrm{sk},3} = \begin{pmatrix} t_{\mathrm{sk},13} & t_{\mathrm{sk},23} \end{pmatrix}', \\ & t_{\mathrm{sk},5} = \begin{pmatrix} t_{\mathrm{sk},15} & t_{\mathrm{sk},25} \end{pmatrix}', \end{aligned} \tag{11}$$

Based on the task decomposition, $t_{\mathrm{sk,ij}}$, the decomposed tasks are scheduled as follows,

$$\begin{aligned} & t_{\mathrm{sch},21} = \begin{pmatrix} s_{\mathrm{tk1},10} & s_{\mathrm{tk1},50} \end{pmatrix}, \\ & s_{\mathrm{tk1},10} = \begin{pmatrix} s_{\mathrm{tk1},11} & s_{\mathrm{tk1},21} & s_{\mathrm{tk1},31} \end{pmatrix}', \\ & s_{\mathrm{tk1},11} = 0, \;\; \text{no assignment}, \\ & s_{\mathrm{tk1},21} = \begin{pmatrix} s_{\mathrm{tk1},3} & s_{\mathrm{tk1},4} \end{pmatrix}, \\ & s_{\mathrm{tk1},31} = \begin{pmatrix} s_{\mathrm{tk1},1} & s_{\mathrm{tk1},2} & s_{\mathrm{tk1},3} & s_{\mathrm{tk1},4} \end{pmatrix}, \end{aligned} \tag{12}$$

$$\begin{aligned} & s_{\mathrm{tk1},50} = \begin{pmatrix} s_{\mathrm{tk1},15} & s_{\mathrm{tk1},25} \end{pmatrix}', \\ & s_{\mathrm{tk1},15} = \begin{pmatrix} s_{\mathrm{tk1},5} & s_{\mathrm{tk1},6} & s_{\mathrm{tk1},7} \end{pmatrix}, \\ & s_{\mathrm{tk1},25} = \begin{pmatrix} s_{\mathrm{tk1},8} & s_{\mathrm{tk1},9} \end{pmatrix}, \end{aligned}$$

Table 3: Decomposed subtasks in Task 23.

| Subtask No. | Subtasks |
|---|---|
| 1 | Iq leadership |
| 2 | No Rp to Iq |
| 3 | Rp Iq/Rv Rp |
| 4 | Rq 2D map/Rv |
| 5 | Rq tgt data/Rv |
| 6 | Tr lnd cmd/ Rv Rp |
| 7 | Merge map/tgt data |
| 8 | Online scheduling |
| 9 | Tr path/Rv Rp |
| 10 | Tr agn/Rv Rp |
| 11 | Tr 2D map/Rv Rp |
| 12 | Tr tgt data/Rv Rp |
| 13 | Tr leadership/Rv Rp |
| 14 | Rv Rp to Iq/Rp |
| 15 | Rv Rq/Tr 2D map |
| 16 | Rv Rq/Tr tgt data |
| 17 | Rv lnd cmd/Rp |
| lnd | landing |
| Iq | inquire/inquiry |
| Rq | request |

$$\begin{aligned} t_{\mathrm{sch},12} &= \begin{pmatrix} s_{\mathrm{tk}2,1} & \cdots & s_{\mathrm{tk}2,\mathrm{n}_{\mathrm{st}2}} \end{pmatrix}, \\ t_{\mathrm{sch},4} &= \begin{pmatrix} s_{\mathrm{tk}4,1} & s_{\mathrm{tk}4,2} \end{pmatrix}, \end{aligned} \tag{13}$$

$$\begin{aligned} & t_{\mathrm{sch},23} = \begin{pmatrix} s_{\mathrm{tk}3,1} & s_{\mathrm{tk}3,20} \end{pmatrix}, \\ & s_{\mathrm{tk}3,20} = \begin{pmatrix} s_{\mathrm{tk}3,120} & s_{\mathrm{tk}3,220} \end{pmatrix}', \\ & s_{\mathrm{tk}3,120} = \begin{pmatrix} s_{\mathrm{tk}3,14} & s_{\mathrm{tk}3,15} & s_{\mathrm{tk}3,16} & s_{\mathrm{tk}3,17} \end{pmatrix}, \\ & s_{\mathrm{tk}3,220} = \begin{pmatrix} s_{\mathrm{tk}3,2} & s_{\mathrm{tk}3,30} & s_{\mathrm{tk}3,7} \\ s_{\mathrm{tk}3,80} & s_{\mathrm{tk}3,110} \end{pmatrix}, \\ & s_{\mathrm{tk}3,30} = \begin{pmatrix} s_{\mathrm{tk}3,130} & s_{\mathrm{tk}3,230} \end{pmatrix}', \\ & s_{\mathrm{tk}3,80} = \begin{pmatrix} s_{\mathrm{tk}3,18} & s_{\mathrm{tk}3,28} \end{pmatrix}', \\ & s_{\mathrm{tk}3,110} = \begin{pmatrix} s_{\mathrm{tk}3,111} & s_{\mathrm{tk}3,211} \end{pmatrix}', \\ & s_{\mathrm{tk}3,18} = 0, \ \text{no assignment}, \\ & s_{\mathrm{tk}3,28} = \begin{pmatrix} s_{\mathrm{tk}3,8} & s_{\mathrm{tk}3,9} & s_{\mathrm{tk}3,10} \end{pmatrix}, \\ & s_{\mathrm{tk}3,\mathrm{k}30} = \begin{pmatrix} s_{\mathrm{tk}3,3} & s_{\mathrm{tk}3,4} & s_{\mathrm{tk}3,5} & s_{\mathrm{tk}3,6} \end{pmatrix}, \\ & s_{\mathrm{tk}3,\mathrm{k}11} = \begin{pmatrix} s_{\mathrm{tk}3,11} & s_{\mathrm{tk}3,12} & s_{\mathrm{tk}3,13} \end{pmatrix}, \\ & k \in \{1,2\}. \end{aligned} \tag{14}$$

Based on the submission/ task schedules, the decomposed tasks/ subtasks can be assigned accordingly. The submission/ task assignment and progress vectors record the completed, being completed and to be completed step/ task/ subtask. The next step/ task/ subtask is clear based on the event activation conditions.

The submission event state transition is shown in Figure 4. The event state transition in Task 21 is shown in Figure 5. There are three branches from the enter for the three batches respectively. The UAVs in the first batch do not need to receive the online scheduled path and assignment, the merged 2D map and targets. The UAVs in the second batch do not need to receive the online scheduled path and assignment, but they need to receive the merged 2D map and targets. The UAVs in the third batch if applicable need to receive the online scheduled path and assignment, as well as the merged 2D map and targets. The subsequent two branches are for the batch leader and members respectively.

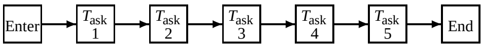


Figure 4: Submission event state transition.

The event state transition in Task 23 is shown in Figure 6. The UAVs in a batch attend the leader selection for the new batch leader. There are two branches of the outcome. One is for the members and the other is for the new batch leader. The leader needs to collect the data from the two members respectively in two branches. After merging the 2D map and found targets, the leader conducts the online schedule only in the second batch. Thus, one branch is no assignment and the other is for the online schedule. Subsequently, the leader needs to hand over the merged map and found targets to the standby UAVs in the next batch in one branch and in the other branch hand over them to the ground station at end of the search. This completes the individual event control design.

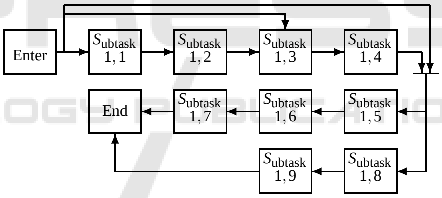


Figure 5: The event state transition in Task 21.

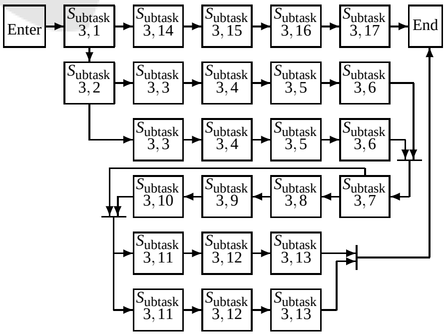


Figure 6: The event state transition in Task 23.

## 3.3 Conversion of Event Commands to Dynamic Commands

Eighteen sets of the knowledge are defined for the message subtasks and two sets of the action are defined for the action subtasks. The flight subtasks are presented as the line segnebts in the starting and end points and thus they need to be converted into the trackable references of the position, velocity and acceleration by the UAVs.

There may be four phases to complete the flight subtasks such as the acceleration (Ac), velocity holding (Hd), deceleration (Dc) and hovering (Hv). The first three phases are velocity tracking and the last phase is position tracking.

$$\begin{cases} H\mathrm{v}, & d \le d_{\mathrm{hv}} \\ Ac + H\mathrm{v}, & d_{\mathrm{hv}} < d \le 2d_{\mathrm{hv}}, \\ Ac + Dc + H\mathrm{v}, & 2d_{\mathrm{hv}} < d \le d_{\mathrm{hd}}, \\ Ac + Hd + Dc + H\mathrm{v}, & d > d_{\mathrm{hd}}, \end{cases} \tag{15}$$

where $d_{\mathrm{hv}}$ denotes the distance that the UAVs can hover from one point to another point. $d_{\mathrm{hd}}$ denotes the distance that the UAVs need to hold the maximal speed to fly. $d_{\mathrm{hd}} = v_{\mathrm{max}}^2 / a_{\mathrm{max}}$ with $v_{\mathrm{max}}$ and $a_{\mathrm{max}}$ being the maximal speed and acceleration. $d$ denotes the distance between the starting point, $p_{\mathrm{s}}$, and end point, $p_{\mathrm{e}}$. $d = \|p_{\mathrm{e}} - p_{\mathrm{s}}\|$. The trackable references can be computed well for the four phases each. The heading reference is the direction of the flight subtask from the starting point to the end point in the first three phases and it is the direction of the next flight subtask in the last phase or pointing to the North if the next flight subtask is hovering. With such conversions, the event commands are executable.

### 3.4 Event-driven Transition

A set of the event activation conditions are defined to control the transition of the discrete event states. The flight subtask completion conditions are defined based on the distance along the direction of the flight subtask. The flight subtask is completed when

$$d_{\mathrm{f}} \le 0, \quad d_{\mathrm{f}} = (p_{\mathrm{e}} - p)'(p_{\mathrm{e}} - p_{\mathrm{s}})/d,$$

where $p$ denotes the position of the UAV. $d_{\mathrm{f}}$ denotes the projection of the distance to be flied relative to the end point to the direction of th flight subtask. If the flight subtask is hovering, it is completed when the hovering is over the given duration. With the flight subtask completion conditions, the transition of the discrete event states is clear.

## 4 SIMULATION

The simulation is conducted to verify the designed AMM system to coordinate nine of the UAVs to search the field of forest. The resulting closed-loop system is shown in Figure 7 in which Gds denotes the ground station, Cmm denotes the communication system and Env denotes the surrounding environment.

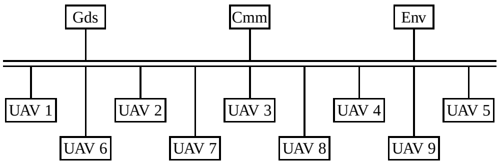


Figure 7: A simulation system of multiple UAVs.

In the simulation,, one UAV is assumed lost in the first batch and another UAV is lost in the second batch. Based on the merged map, the online schedule decides to assign two UAVs and schedules the path for them each to search the missed areas in the third batch. Then, the full of the forest field is searched.

The discrete event states and flight trajectories of the UAVs are shown in Figure 8. The 2D maps built and merged by the UAVs to describe the detected areas are shown in Figure 9. The simulation results demonstrate that the designed AMM system is successful to coordinate the group of UAVs to search the field of forest together. The number of the defined events is not too many to affect the implementation of the AMM system. The designed AMM system is also successfully verified in our high-fidelity simulator.

## 5 CONCLUDING REMARKS

The enhanced HDM has been successfully applied to design an AMM system to collaborate multiple UAVs to complete a designated mission together. The main features of the designed AMM system are hierarchical control, series and/or parallel decomposition and distributed implementation. The missions can be decomposed perfectly with the series and/or parallel relationships in multiple levels. The enhanced HDM is applicable to the other more complex scenarios. Nonetheless, the mission reschedule is to be studied when an accident happens.

## REFERENCES

Bellingham, J., Tillerson, M., Alighanbari, M., and How, J. (2002). Cooperative path planning for multiple uavs in dynamic and uncertain environments. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 2816–2822, Las Vegas, Nevada, USA. IEEE.

Inalhan, G., Stipanovic, D., and Tomlin, C. (2002). Decentralized optimization with application to multiple aircraft coordination. In *Proceedings of the 41st IEEE InternationalConference on Decision and Con-*

*trol*, pages 1147–1155, Las Vegas, Venada, USA. IEEE.

Jadbabaic, A., Lin, J., and Morse, A. S. (2003). Coordination of groups of mobile autonomous agents with neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):998–1001.

Kaminer, I., Yakimenko, O., Pascoal, A., and Ghabcheloo, R. (2006). Path generation, path following and coordinated control for timecritical missions of multiple uavs. In *Proceedings of 2006 American Control Conference*, Minneapolis, MN, USA. IEEE.

Kopeikin, A. N., Ponda, S. S., Johnson, L. B., and How, J. P. (2013). Dynamic mission planning for communication control in multiple unmanned aircraft teams. *Unmanned Systems*, 1(1):41–58.

Lafferriere, G., Williams, A., Caughman, J., and Veerman, J. E. (2005). Decentralized control of vehicle formations. *Systems and Control Letters*, 54(9):899–910.

Murata, T. (1989). Petri nets: properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580.

Peng, K., Pang, T., Lin, F., and Chen, B. M. (2014). Autonomous mission execution for multiple unmanned aerial vehicles with hierarchical-distributed methodology. In *Proceedings of 2014 11th IEEE InternationalConference on Control and Automation*, pages 1369–1374, Taichung, Taiwan. IEEE.

Richards, A., abd M. Tillerson, J. B., and How, J. B. (2002). Coordination and control of multuple uavs. Aiaa paper 2002–4588.

Teo, R., Jang, J. S., and Tomlin, C. J. (2004). Automated multiple uav flight the stanford dragonfly uav program. In *Proceedings of the 43rd IEEE InternationalConference on Decision and Control*, pages 4268–4273, Atlantis, Geogia, USA. IEEE.

Tomlin, C. J., Lygeros, J., and Sastry, S. S. (2000). A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–969.

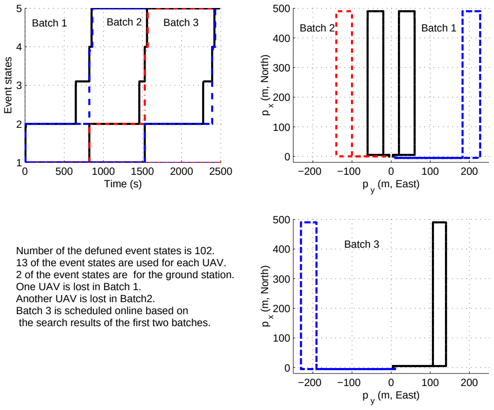
Uhrmann, J. and Schulte, A. (2011). Task-based guidance of multiple uav using cognitive automation. In *Proceedings of 2011 International Conference on Advanced Cognitive Technologies and Applications*, Rome, Utaly. ABIA.

van der Schaft, A. J. and Schumacher, H. (1998). Complementarity modeling of hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):483–490.

Wong-Toi, H. (1997). The synthesis of controllers for linear hybrid automata. In *Proceedings of the 1997 IEEE Conference on Decision and Control*, pages 4607–4613, San Diego, CA, USA. IEEE.

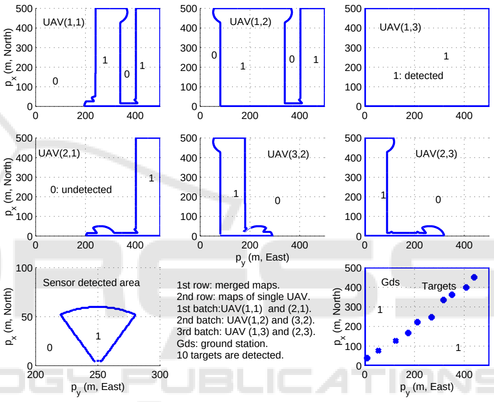
Ye, H., Michel, A., and Hou, L. (1998). Stability theory for hybrid dynamical systems. *IEEE Transactions on Automatic Control*, 43(4):461–474.

# APPENDIX



Number of the defuned event states is 102.
13 of the event states are used for each UAV.
2 of the event states are for the ground station.
One UAV is lost in Batch 1.
Another UAV is lost in Batch2.
Batch 3 is scheduled online based on
the search results of the first two batches.

Figure 8: Event states and flight trajectories of UAVs.



1st row: merged maps.
2nd row: maps of single UAV.
1st batch:UAV(1,1) and (2,1).
2nd batch: UAV(1,2) and (3,2).
3rd batch: UAV (1,3) and (2,3).
Gds: ground station.
10 targets are detected.

Figure 9: Maps of the detected areas by UAVs.