Original Article

Institution of MECHANICAL ENGINEERS

Journal of SYSTEMS AND CONTROL ENGINEERING

Proc IMechE Part I:
J Systems and Control Engineering
2023, Vol. 237(10) 1765–1776

DOI: 10.1177/09596518231173765
journals.sagepub.com/home/pii
Sage

Leader–follower control of multi-unmanned aerial vehicle based on supervisory control theory for a broad tributary area mapping scenario

Jaehwi Seol[1,2], Chanyoung Ju[1,2] and Hyoung Il Son[1,2]

Abstract

This study proposes a discrete event system-based control strategy for autonomous tributary mapping using multi-unmanned aerial vehicle. When considering the unmanned aerial vehicles as discrete event systems, supervisory control theory is used to model and control individual unmanned aerial vehicle behavior in the system. In tributary mapping, the situation changes each time depending on environmental factors (e.g. weather) and the work must be performed in an unstructured environment. Therefore, this article proposes a multi-unmanned aerial vehicle-based supervisory control system to solve real-field problems. Unlike the control systems of recent studies, which mainly deal with continuous-time dynamics, we modeled a multi-unmanned aerial vehicle system based on a discrete event system in which the dynamic states are transitioned by asynchronous events. The proposed multi-unmanned aerial vehicle-based supervisory control system was validated in dynamic simulators and demonstrated that multi-unmanned aerial vehicle satisfies the behavior specifications. The supervisor proposed in this study was validated using a physics-based simulator.

Keywords

Tributary mapping, supervisory control, leader–follower control, modular supervisor, multi-unmanned aerial vehicle

Date received: 25 March 2022; accepted: 10 April 2023

Introduction

Recently, with the increase in environmental pollution, the need for management systems that can effectively manage natural systems and ecosystems has emerged.[1–4] Furthermore, in agriculture, a system to minimize environmental damage and manage it is being studied.[5,6] Among the environmental management systems, data obtained from satellite maps of the natural environment of tributaries are used for water quality management. However, satellite-based mapping has limitations, such as high cost and low availability, depending on the weather.[7] Furthermore, it is difficult to obtain accurate information, such as the width of a branch, because of the low resolution. Direct observation by humans can be used to obtain accurate information; however, it is highly inefficient because of the vast area of the tributary. Therefore, a system that can effectively obtain information on tributaries is required to manage water quality efficiently and better understand the impact of global environmental changes on ecosystems.[8]

Several studies have been conducted to perform unmanned aerial vehicle (UAV)-based tributary mapping to acquire tributary information.[9,10] Compared with the tributary mapping of the existing satellites, UAVs can fly from a lower altitude to a higher altitude, less interference from ground obstacles.[11,12] Moreover, high-resolution images can be obtained from tributaries, enabling more precise tributary management.

[1]Department of Convergence Biosystems Engineering, Chonnam National University, Gwangju, Republic of Korea
[2]Interdisciplinary Program in IT-Bio Convergence System, Chonnam National University, Gwangju, Republic of Korea

* Chanyoung Ju is also affiliated to Automotive Materials & Components R&D Group, Korea Institute of Industrial Technology, Gwangju, Republic of Korea

Corresponding author:
Hyoung Il Son, Department of Convergence Biosystems Engineering and Interdisciplinary Program in IT-Bio Convergence System, Chonnam National University, Yongbong-ro 77, Gwangju 61186, Republic of Korea.
Email: hison@jnu.ac.kr

In previous studies, UAV-based mapping for managing natural systems has been performed using a single UAV.[13] A single UAV has clear limitations in covering unstructured and large-scale natural systems. In particular, single UAV-based mapping requires a longer exploration time, and as the exploration range expands, its limitations become more prominent.[14] Therefore, many studies on multi-UAV systems are being conducted to overcome the limitations of single UAVs.[15–21]

This study addresses the limitations of single UAV-based mapping in previous studies. The limitations of a single UAV are real-world practical problems with low coverage area and long exploration time. In addition, single UAV is inefficient and cannot cover tributary, which is an unknown environment with a random split. These problems can be solved using multiple UAVs. However, because the control complexity increases, a control method that considers the mapping scenario is required. We solve these problems by proposing a supervisory control theory (SCT)-based multi-UAV system for tributary mapping. Instead of modeling as a continuous-timed system to solve the existing control complexity, we solve the problem by modeling as a discrete event system (DES).

Therefore, this study proposes a leader–follower approach based on supervisory controllers[22] to extend to multiple UAVs to overcome the limitations of single UAV-based systems and solve the existing control complexity. The proposed approach can be used in practical applications for tributary mapping, in which the concept of leader–follower control is applied to change the formation according to the mapping scenario. If the tributary reaches a split tributary, the proposed system recreates and controls the trajectory and groups of each UAV. This study contributes to how to define the tributary mapping problem, how to model multi-UAV systems based on formal methods, how to design motion specifications for cooperative control, and how to validate them.

Related works

In the work by Nuske et al.,[9] UAV-based exploration was developed to map rivers without GPS and pre-mapping. The UAV operated beneath the tree line and performed peripheral three-dimensional (3D) obstacle-based path planning with onboard sensing to navigate the river. In the work by Jiang et al.,[23] hydrodynamic models of rivers and streams were constructed leveraging water level or water surface elevation (WSE) observations to increase the model reliability and predictive level. A radar altimeter system was used on a UAV to simulate and map the spatially distributed WSE. Autonomous river mapping was proposed for the autonomous exploration of rivers without any prior information using onboard control to perceive rivers, banks, and obstacles.[9,24] That study provided 3D maps and verified them against the guidance decisions made by a human piloting a boat carrying the proposed system over several kilometers.

Numerous studies have been conducted to expand to a multi-UAV system and overcome the limitation of a single UAV.[19,21] Multi-UAV systems predominantly use cooperative and large-scale controls to determine the team behavior of multiple autonomous vehicles.[25] The formation control of a multi-UAV system aims to meet prescribed constraints on the state by driving multiple UAVs, usually by applying behavior-based,[26] leader–follower,[27–29] and graph theory-based[30] methods. In the work by Mwaffo et al.,[29] the authors explicitly modeled the presence of stochastic perturbations affecting the dynamics and developed multi-vehicle systems using stochastic differential equations. Based on modeling, the authors proposed a decentralized approach to the large-scale control of stochastic mobile UAV groups.

Cai and Wonham[31] proposed SCT to control under a DES. Supervisory control-based approaches for DESs have been used to control multi-UAV systems.[32] In the work by Lopes et al.,[33] an SCT-based controller was proposed for swarm UAVs, and the performance of the proposed controller was verified through an experiment clustering a group of UAVs based on open-source software. In the work by Ju and Son,[34] the authors proposed a hybrid system that combined a continuous-time system and DES for large-scale dynamic systems, such as field UAVs. Their research expanded on a previous study[35,36] by considering partial observation caused by the inability to monitor events using sensors because of noise, disturbances, and failure.

Contribution

The contributions and novelty of this study can be summarized as follows:

1. We expand leader–follower formation control-based multi-UAV system from a tributary mapping perspective. Our approach overcomes the limitations of the single UAV as a simple method using leader–follower formation control.
2. We propose an SCT-based multi-UAV system for scalability from a control perspective. The SCT-based modeling and control system has the advantage of scalability.
3. We conducted an experiment in a virtual environment similar to the realistic environment of a tributary with three split tributaries. The proposed supervisory controllers were fully implemented in a physic-based robot simulator.

Structure of article

The organization of this article is given as follows. In section "Preliminaries," we introduce the main problem in tributary mapping and briefly review the concepts of DESs and SCT. In section "System modeling," we propose a supervisory controller with certain specification based on the SCT. In section "Results," we describe the implementation of the proposed supervisor and the verification of the simulation and results, and finally, we present our conclusions.

Preliminaries

The problems and notations of the SCT used in this study are summarized in this section.

Problem description

Tributaries have different types of characteristics. There are various types, such as brush-covered, small range, and broad tributaries. Because the state of a tributary is affected primarily by environmental phenomena, when a drought or flood occurs, it becomes difficult to manage the map because the current shape of the current tributary is distorted compared with the previously created map. As the environment changes, it is necessary to periodically attempt tributary mapping to manage the tributaries precisely and effectively.

A single UAV requires multiple flights to perform periodic tributary mapping. Because large tributaries have a vast range, a single UAV mapping scenario is limited by its batteries. Furthermore, it is unknown when a broad tributary will diverge into a branching tributary. Mapping should be performed by selecting a path for the bifurcated tributary, as depicted in Figure 1. This approach is considered inefficient because it is time-consuming and expensive, requiring multiple flights to map every tributary.

It is necessary to completely map all tributaries in the shortest time. Therefore, we expand from a single UAV to a multi-UAV framework and cover tributaries including large tributaries and multiple sub-tributaries to achieve these goals. The branch depicted in Figure 1 can be split into several different stems. Consider the following situation to navigate its tributary quickly and with minimal flight.

The mapping task covers the scenario of mapping a tributary via a UAV equipped with a gimbal camera. For split tributaries, multi-UAVs are used to navigate all areas of the tributary. We assumed that the sensor is always aimed at a straight line, even though the UAV performs a rolling maneuver, and the river is considered a curve under the UAV's physical constraints. Furthermore, obstacles present in the tributary mapping are not considered, and it is assumed that no obstacles interference occurs during the flight.

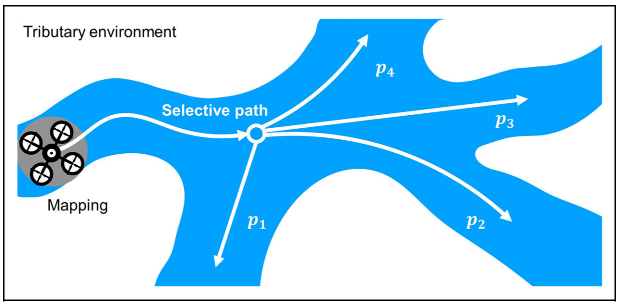

Figure 1. Schematic of tributary mapping using a UAV when the tributary splits into branches. The UAV should explore all the zones for tributary mapping.

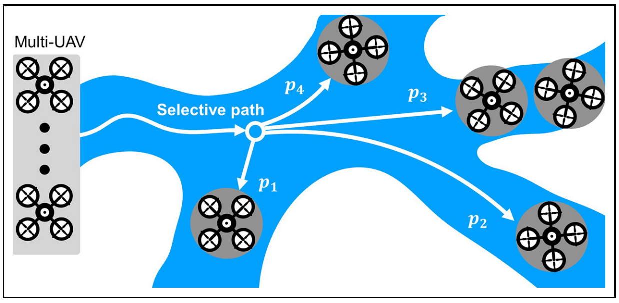

Figure 2. In a multi-UAV system, the control strategy for navigating all tributaries is to move the frontmost UAV to the nearest branch.

The multi-UAV control system must determine the trajectory of each UAV to navigate all tributary branches, as depicted in Figure 2. As a way to effectively navigate all areas, multiple UAVs must visit their branching tributaries in all areas. In this case, the multi-UAV uses the leader–follower formation control method. In practice, the tributary splits unexpectedly during operation. The existing leader–follower formation follows the group's leader; in this case, an unrecoverable problem occurs when the branch is divided. Therefore, a system that can map all diverging branches can be developed by enhancing the control law elaborate by subdividing the group.

Remark 1. Our key idea is that if the leader UAV splits all the branched tributaries, then the multi-UAV system can visit all the split branches. As the flight continues, small tributaries are created, but the more the number of UAVs, the more accurate mapping is possible. Furthermore, if the number of UAVs is insufficient to cover all the tributaries, it can at least cover the divergent area depending on the altitude of the UAV; it is unnecessary to perform precise mapping for tributaries that are too small.

Desired behavior of multi-UAV system

The desired behavior of a multi-UAV system is illustrated in Figure 2. First, multi-UAVs form a group by determining the leader and follower UAVs. Then, the group is divided into the main group (i.e. include leader UAV and follower UAV) and subgroup (i.e. follower UAVs). The leader–follower formation topology and the group and follower UAVs, as shown in Figure 3, are used to determine the leader of the subgroup. In the split zone, the leader UAV selects a branch, such as a narrow tributary, and moves, while the follower UAVs move to other tributaries.

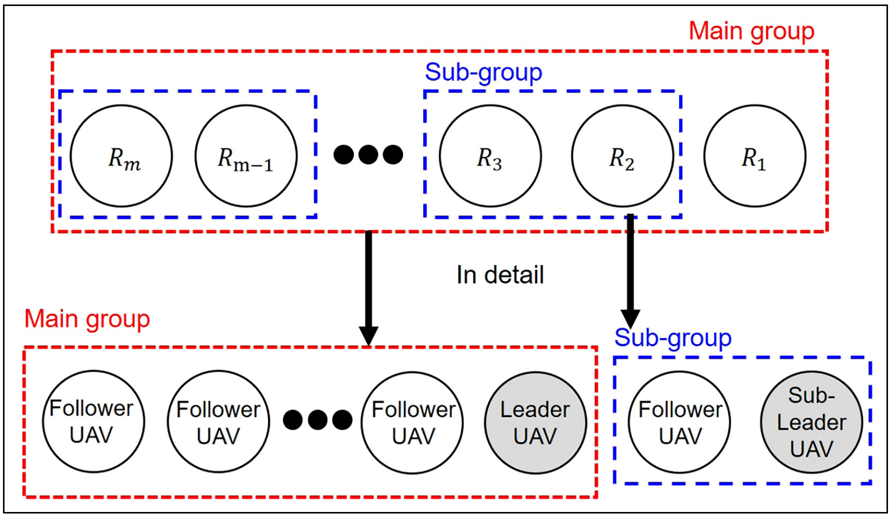

Figure 3. Multi-UAV formation topology: the desired behavior in a multi-UAV group, divided into a main group and subgroup. The main group consists of the main leader UAV (R1) and other UAVs (follower UAVs). The subgroup consists of the sub-leader and follower UAVs.

The subgroup of the sub-leader UAV is with loss of generality taken as the follower UAV. The sub-leader follows the main leader as a follower UAV; when the main leader is lost, a new main leader must be assigned. Among the sub-leader UAVs, the main leader UAV is selected from the group containing the main leader.

Remark 2. The motion control of each UAV is regrouped by a distributed supervisor when a leader UAV is lost. This control method can be extended to an infinite number of multi-UAVs.

The groups thus created achieve the desired specific formation as specified in the desired distance d_{ij}, a control input to avoid collisions between UAVs. The formation maintains a formation that maintains relative distance in a leader–follower control. A control law for the UAV i, $u_i^c \in \Re^3 \quad \forall i = 1, \ldots, N, \text{and } \forall j \in \mathcal{N}_i$ as defined by

$$u_i^c := \sum_{j \in \mathcal{N}_i} \partial_{ij} \left(\| p_i - p_j \|^2 \right)^T \tag{1}$$

where $\mathcal{N}_i$ is the dynamic neighbor set of ith UAV, p_i is the position vector of UAV i, and ∂_{ij} obtains information from the neighbor UAV j through a network. First, the desired relative distance $d_{ij}^c = \| p_i - p_j \|$ is set between the UAV j and the leader in the vertical direction. ∂_{ij} is a designed potential function that produces an attractive behavior if attractive action $\| p_i - p_j \| > d_{ij}^c$ and a repulsive action if $\| p_i - p_j \| < d_{ij}^c$.

Supervisory control theory

Here, we briefly overview the SCT and DES in this section; please refer to Cai and Wonham[31] for further details. The SCT was developed to provide a formal methodology for the automatic synthesis of DES controllers.

Definition 1. A finite-state automaton G is defined as a five tuple consisting of the following elements[37]

$$\mathrm{G} = (\mathrm{Q}, \Sigma, \delta, \mathrm{q}_0, \mathrm{Q}_\mathrm{m}) \tag{2}$$

where Q is the set of all states, Σ is the set of all events, δ is the state transition function of G ($\delta : \mathrm{Q} \times \Sigma^* \to \mathrm{Q}$), q_0 is the initial state of G, and Q_m is the subset of marker states, which indicates a desired state $\mathrm{Q}_m \subseteq \mathrm{X}$. In the transition function δ, Σ^* represents a string of events containing the null event ϵ. The event set is partitioned into a set of controllable events Σ_c and a set of uncontrollable events $\Sigma_{uc} (\Sigma = \Sigma_c \cup \Sigma_{uc})$.

Definition 2. The closed behavior of G is the language as follows

$$\mathrm{L(G)} = \mathrm{s} \in \Sigma^* | \delta(q_0, s)! \tag{3}$$

where $\delta(q_0, s)!$ indicates that the next state in which the string s occurs at q_0 is defined in G. Furthermore, the marked behavior of G is defined as $L_m(G) := \{ s \in L_m(G) | \xi(x_0, s) \in Q_m \}$. A string s is a prefix of the string t, and the prefix closure of language L is stated as $\bar{L} = \{ s \in \Sigma^* | (\exists t \in \Sigma^*)[st \in L] \}$

Definition 3. The marked behavior of a language L, denoted as $\mathrm{L_m}$, is defined as follows

$$\mathrm{L_m(G)} = \{ \mathrm{s} \in \mathrm{L(G)} : \delta(\mathrm{q}_0, \mathrm{s}) \in \mathrm{Q_m} \} \subseteq \mathrm{L(G)} \tag{4}$$

Table 1. Events for each UAVs.

State	Event	Description	Controllable
General	α_1	Arming	○
	α_3	Disarming	○
	α_5	Take-off	○
	α_7	Landing	○
	α_9	Hovering	○
	α_{11}	Move to tributary	○
Leader	β_1	Start mission	○
	β_3	Move to narrow tributary	○
	β_5	Send message to follower UAV	○
	β_7	Return to home	○
	β_9	Altitude up	○
	β_{11}	Altitude down	○
	β_2	Finish mission	○
	β_4	Tributary detected	×
	β_6	Tributary not detected	×
	β_8	Tributary detected more than two	×
Follower	γ_1	Following leader UAV	○
	γ_2	Lost leader	×
	γ_3	Following sub-leader UAV	○
	γ_5	Received message from leader UAV	○
Assign	σ_1	Assigned sub-leader UAV	○
	σ_2	Assigned leader UAV	○
	σ_3	Assigned follower UAV	○

UAV: unmanned aerial vehicle.

where G is nonblocking if $\overline{L_m(G)} = L(G)$. The marked state can be reached after a string occurs in any state of G. The nonblocking is defined as reachability because the appropriate automaton G and its language L(G) can fall into deadlocks or livelocks when L(G) is blocking.

Plant G and its behavior specification E are modeled as automata $G = (X, \Sigma, \xi, \alpha, X_0)$ and $E = (Q, \Sigma, \delta, q_0)$. The supervisor is also defined as an automata $S = \{Y, \Sigma, \xi, \beta, Y_0\}$.

Definition 4. Controllability is defined as the following condition being satisfied

$$(\forall s, u) \in \Sigma_{uc}, su \in L(G) \rightarrow su \in \overline{L(G)} \tag{5}$$

where u is an uncontrollable event.

The language $K \subseteq \Sigma^*$ is controllable (w.r.t. G) provided for all $s \in \bar{K}$. A supremal (i.e. the maximum) element exists and is denoted by $supC(K)$. Specification language E is proposed to restrict the behavior of the plant. The specification language $E \subseteq \Sigma^*$, and E has an optimal (maximally permissive) supervisor S.

System modeling

Plant modeling for multi-UAV system

We proposed an approach to tributary navigation using multiple UAVs with leader–follower formation control for tributary mapping. Therefore, one of the goals of this study was to enable collaboration between field UAVs to design automata models for tributary mapping consisting of leader and follower UAVs. Automata models represent the initial states, double lines indicate marked states, and the events and state transition functions are specified by alphabetical designations and arrows. Table 1 lists the event, description, and status as controllable or uncontrollable. Moreover, all events are categorized by state, indicating that an event can change at any time with a state change. Odd-numbered events are controllable events (indicated by ◯), and even-numbered events are uncontrollable events (indicated by ×). All events are assumed to be observable.

In a multi-UAV system modeled as a DES, the plant consists of n sub-plants (UAVs), where each UAV is modeled by a finite-state automaton $G = \{X, \Sigma, \xi, x_0, X_m\}$. Sub-plant models are categorized according to the leader–follower formation control strategy, as shown in Figure 4. In addition, the sub-plant is designed to include the assign UAV to consider a tributary mapping scenario because the leader and follower not only initially assign roles but also have to reassign them according to the environment. Therefore, the overall plant G is defined as the parallel composition of the n component UAVs $G = (U, \gamma, \xi, u_0, U_m)$. In this study, the plant model G is the synchronous product of n component UAVs (Definition 4), language G consists of a finite-state automaton, leader UAV G_L, follower UAV G_F, and assign UAV state G_A. Plant model G is obtained as $G = G_L||G_F||G_A$. Each sub-plant model includes three states: general state, follower UAV state, and leader UAV state.

Definition 5. The overall plant G is obtained by the automaton synthesized by the following operation

$$Q = (Q_1 \times Q_2, \Sigma = \Sigma_1 \cup \Sigma_2, q_0 = (q_{0,1}, q_{0,2}), Q_m = Q_{m,1} \times Q_{m,2}, (q_1), q_2) \in Q_1 \times Q_2, \forall s \in \Sigma_1 \cup \Sigma_2 \tag{6}$$

where || denotes the synchronous product of languages and Q is synchronous with $Q_1||Q_2$ According to Definition 5, the overall plant is obtained by $G = G_1||G_2||\ldots||G_n$.

The obtained overall plant G consists of 15 states, 67 transitions, and 23 events. Here, this plant is obtained by the leader UAV, follower UAV, and assign UAV. This plant includes one leader UAV, one follower UAV, and one assign UAV. As the number of UAVs increases, the overall plant scale increases rapidly. Owing to the nature of the tributary, which is not

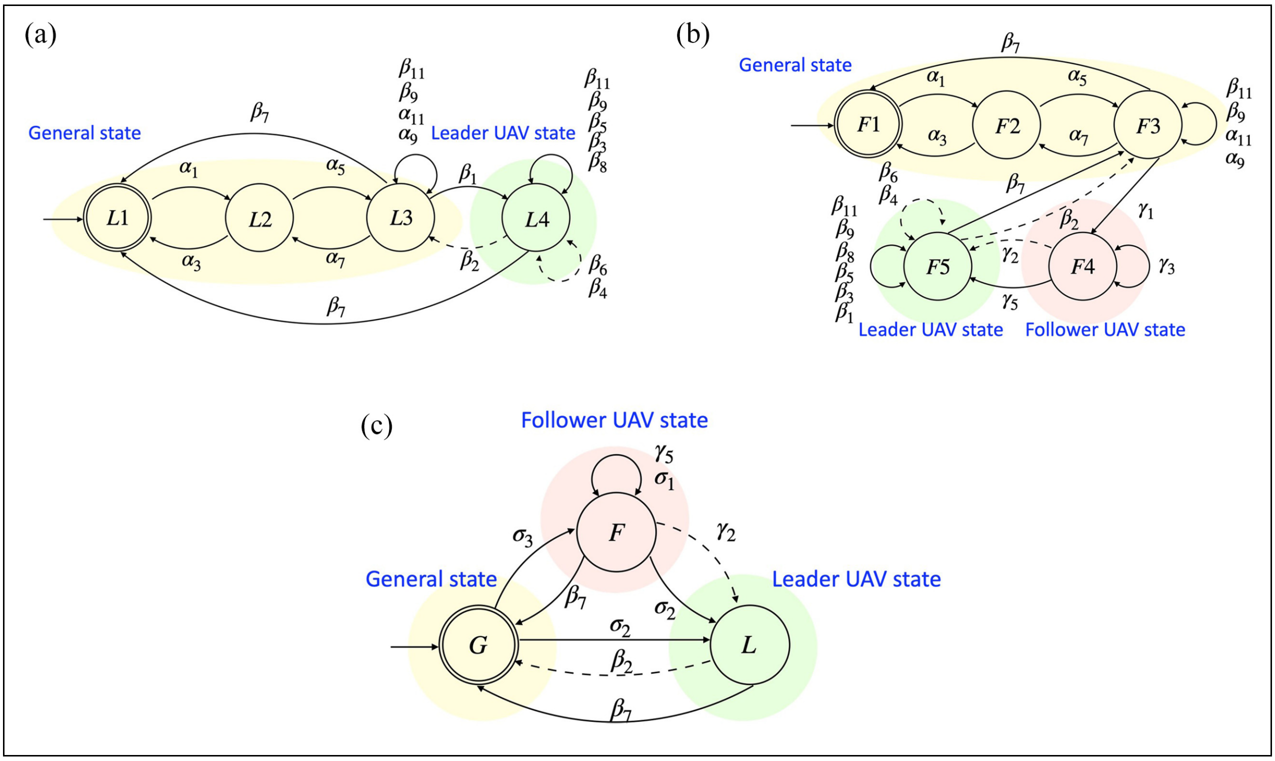

Figure 4. Automata model for each sub-plant: (a) leader UAV G_L, (b) follower UAV G_F, and (c) assign UAV state G_A.

constant, it is not possible to accurately predict the split zone; therefore a large number of UAVs are required. Consequently, the tributary mapping system is composed of a large-scale plant.

Leader UAV model. The automaton of leader UAVs model, G_L (shown in Figure 4(a)), represents the states and events of the UAVs. The states are defined as $G_L^i = \{X_L^i, \Sigma_L^i, \xi_L^i, x_{L,0}^i, X_{L,m}^i\}$, for $i \in \{1, 2, ..., n\}$. The states are $G_L = \{L_1, L_2, L_3, L_4\}$, where L_1: Ideal, L_2: Arming, L_3: Hovering, and L_4: Performing task. The leader UAV model consists of the general state and leader UAV state. Here, L_1, L_2, and L_3 represent the general state, a set of states before the event (β_1, Start mission). For example, if an event β_1 occurs in state L_3, it transitions from the general state to the leader UAV state. Because all UAVs have a general state, multiple leader UAVs can be created, which makes it possible to search for split tributaries. If β_2 or β_5 occurs in state L_4, it transitions from the leader UAV state to the general state.

Follower UAV model. The automaton of the follower UAV model, G_F (shown in Figure 4(b)), represents the states and events of the UAVs. The states are defined as $G_F^j = \{X_F^j, \Sigma_F^j, \xi_F^j, x_{F,0}^j, X_{F,m}^j\}$, for $j \in \{1, 2, ..., m\}$. The states are $G_F = \{F_1, F_2, F_3, F_4, F_5\}$, where F_1: Ideal, F_2: Arming, F_3: Hovering, F_4: Following leader UAV, and F_5: Performing task. The follower UAV model includes the general state, leader UAV state, and follower UAV state. The follower UAVs model includes the leader UAV state because the leader UAV has to selectively map when it detects the split zone. Furthermore, follower UAVs may lose leader UAVs (e.g. communication and battery); thus, they cannot map the desired explored area. In this case, the follower UAV should be transitioned into a leader UAV to cover the split tributary. Therefore, the follower UAV model consists of the general state, follower UAV state, and leader UAV state. The general state occurs before the assignment of the follower UAV. Here, the follower UAV states F_1, F_2, and F_3 depict the general state, F_4 the follower state, and F_5 the leader state.

Assign model. The automaton of the assigned UAV model, G_A (shown in Figure 4(c)), represents the states and events of the UAVs. Before leader–follower formation control, UAVs are allocated states for leader–follower. The desired behavior of an assign UAV and its states are defined as follows: $G_A = \{G, F, L\}$, where G: general state, F: follower UAV state, and L: leader UAV state. The G_A allocates a UAV state to each UAV, meaning that each UAV state is determined as the general state. The general state is defined before the allocation of leader or follower roles.

Remark 3. In plant modeling, a common state that exists before the leader and follower is assigned. A follower is not only transitioned from a specific state to the leader state but can also become a leader even in

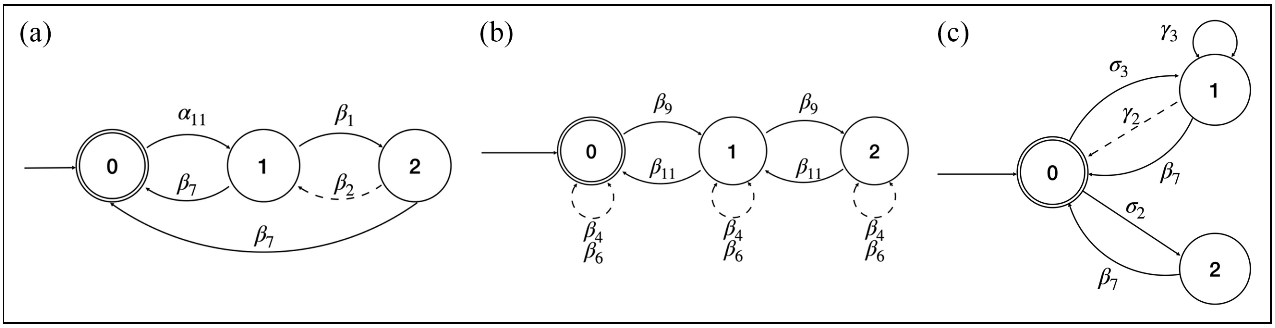

Figure 5. Specification for modular supervisor: (a) Mission performing (SPEC1), (b) Altitude control (SPEC2), and (c) Grouping (SPEC3).

unexpected situations. The follower can transition to the general state and then to the leader state at any time.

Specification modeling

Desired behavior is the specification (i.e. language (K, K_m)) that defines the legal behavior of G under control. We designed a controllable language that meets the conditions for becoming appropriate specifications to satisfy the nonblocking and controllability. The UAV's behavior is implemented based on the specification. For the supervisor, the specifications are defined as follows:

- **SPEC1:** Mission performing: Prior to performing the mission, in the hovering state, the leader UAV moves to the tributary and then performs the mission.
- **SPEC2:** Altitude control: During the mission, the leader UAV can increase altitude if the entire tributary is not sensed.
- **SPEC3:** Grouping: The group is subdivided into a main group and a subgroup. Each subgroup designates its leader, and the follower UAV follows the sub-leader. If the UAV loses its leader UAV, it attempts regrouping.

The designed specifications (Figure 5) are divided according to the roles of the leader and follower. Each UAV is designed to serve a role even when its status changes. SPEC1 and SPEC2 are imposed on the leader UAV and SPEC3 is imposed on every UAV. The mission can be performed more effectively, even in unexpected situations, using the control SPEC3. SPEC3 satisfies all sub-specifications SE1, SE2, and SE3, where SE1 requires that multi-UAVs are divided into a main group and subgroups, SE2 ensures that the subgroups decide the sub-leader, and the follower UAV follows the sub-leader, and SE3 ensures that when the leader UAV is lost, regrouping is initiated. Here, SPEC3 is obtained as $SE1 \wedge SE2 \wedge SE3$.

Synthesis of supervisor S

For plant G and the imposed specification E, let the supervisor be denoted as S. In this study, supervisor S is not an observation supervisor, and we assume that all events are observable. Therefore, the supervisor prevents a non-reachable state in a deadlock loop, disabled event w.r.t. controllability.

For multi-UAV systems, the maintenance of formation between groups involves a series of modular supervisors rather than a single monolithic supervisor. Each modular supervisor maintains a subset of specifications or acts on specific plant components in particular subsets.

Remark 4. Plant G comprises sub-plants G_L, G_F, G_A, and the specification E comprises SPEC1, SPEC2, and SPEC3. The supervisor comprises multiple modular supervisors (S_i), one for each specification, considering the plant. The modular supervisor is computed as $S_i = SupC(E_i, G)$. Each modular supervisor S_k for $k \in \{1, ..., n\}$ is obtained without information about the other supervisors. Modular supervisors can be executed in parallel.

The modular supervisor is obtained using TCT software. For modular supervisory control, the alphabet of E should be equal to the alphabet of G. Therefore, the specification can be satisfied by a self-loop automaton that contain all events in the alphabet of G. In the TCT software, the specifications are designed as follows: $EVENTS = allevent(PLANT)$ and self-loop for all events $E1 = sync(E1, EVENTS)$. Modular supervisor S_1 is obtained as a supervisory controller of the sub-plant by synthesizing $E1$, $S_1 = supcon(PLANT, E1)$.

The plant of behavior that satisfies the specification as shown in Figure 6. The obtained S_1 contains the 23 states, 23 events, and 107 transitions. Similarly, S_2 synthesizes $E2$ that contains 24 states, 23 events, and 84 transitions. S_3 synthesizes $E3$ to obtain a modular supervisor that contains 45 states, 23 events, and 212 transitions. The modular supervisor allows all events that satisfy the control specifications to reach their

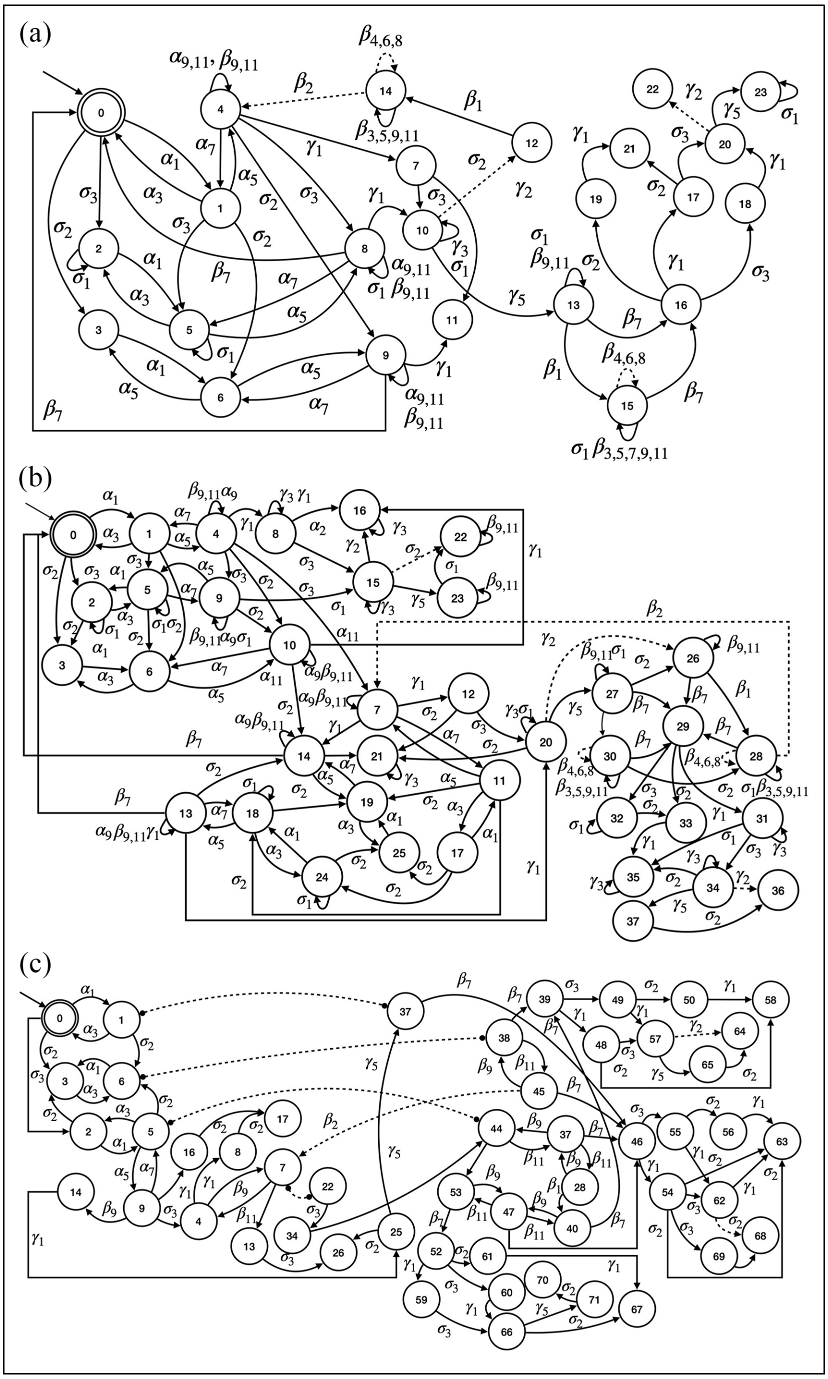

Figure 6. Modular supervisor according to specification: (a) modular supervisor 1 (S_1), (b) modular supervisor 2 (S_2), and (c) modular supervisor 3 (S_3).

maximum. For example, for modular supervisor 3 (S_3), when the plant behaves as a flow of $q_1 = \delta(q_0, \alpha_1)$, $q_4 = \delta(q_1, \alpha_5)$, $q_7 = \delta(q_4, \gamma_1)$, and $q_1 1 = \delta(q_7, \sigma_1)$ from the initial state q_0, it become deadlock in q_{11}. The modular supervisor disables the event for σ_1 in q_7, so that q_{11} is not reached. States after transition from q_{10} to q_{13} cannot reach the marked state. Therefore, the modular supervisor can always transition from q_{10} to q_{13} to the marked state by disabling γ_5.

Results

Experimental setup

In this study, we performed an experiment using a physics-based simulator to validate the proposed supervisory controller for multi-UAV-based tributary mapping. The simulation environment is illustrated in Figure 7. The simulation included five UAVs, comprising one leader UAV and four follower UAVs, and the

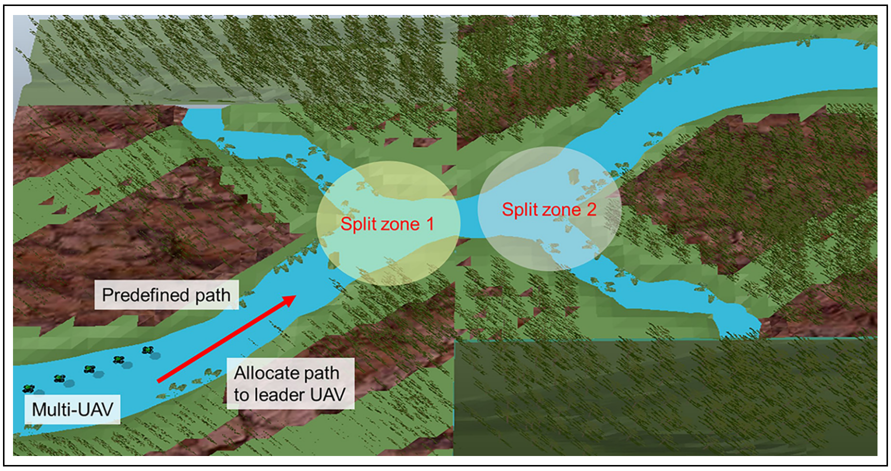

Figure 7. Experimental environment for supervisory control.

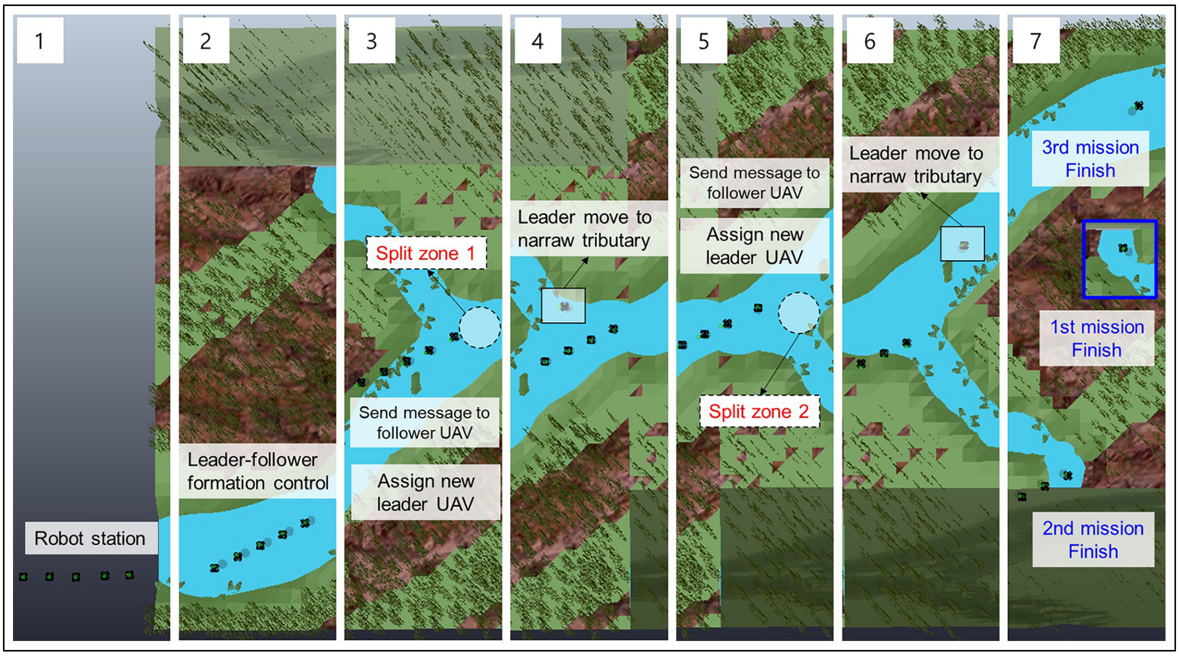

Figure 8. Experimental scenes for multi-UAV system using leader–follower formation control for tributary mapping.

environment was set similarly to a river environment. The tributary split into three sub-tributaries and had two split zones. A physics-based simulator was built to implement and validate a tributary mapping system with multiple UAVs, and a modular supervisory controller was designed for these split zones. Each UAV is equipped with a vision sensor that can recognize a branch, and the designed supervisor can determine the path of each UAV. The system was implemented using CoppeliaSim, a UAV simulator, while low-level and high-level controllers were implemented in MATLAB. The signal was recorded at 50 Hz to evaluate the performance of the proposed supervisory controller. We designed low level and supervisory controllers to control a target plant G modeled by automata. The supervisor provided a feedback control loop that satisfied the behavioral specifications that reflected the control objectives.

Results and discussion

Figure 8 demonstrates scenes from the dynamic progress of the multi-UAV mapping during the experiments. The multi-UAVs are assigned the leader and follower states before the simulation. When assigning a leader and follower UAV, the UAV with the closest relative distance to the leader UAV among the follower UAVs is determined as the sub-leader; then, a group is created among the follower UAVs. In the simulation environment, a path is predefined from the start to end points and assigned to the leader UAV under the assumption that the tributary is recognized without

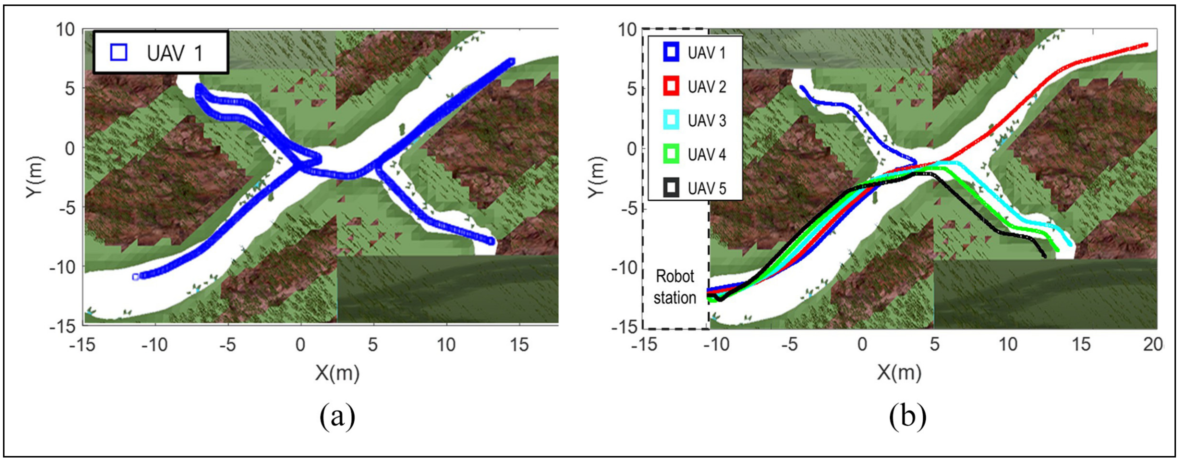

Figure 9. Experimental results of trajectory (a) single UAV and (b) multi-UAV.

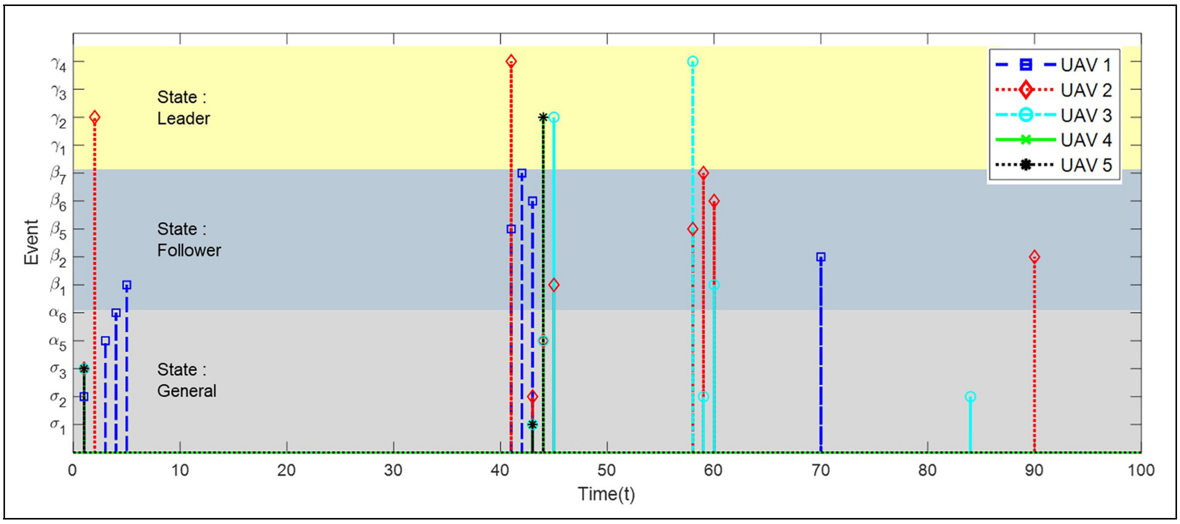

Figure 10. Experimental result of each UAV event sequence.

image processing. When the leader UAV moves along the assigned path, the follower UAVs maintain their relative distance-based formation. Figure 9 illustrates the trajectory of single and multi-UAV in a tributary environment, including a split zone.

SPEC1 and SPEC3, the behavior specifications designed to achieve the control objective, are represented in the simulation. SPEC2 pertains to tributary recognition. Because tributary recognition was not performed in the simulation, the content for altitude control was not implemented. During the simulation, when the *tributary detected more than two* events (β_5) occur, the leader UAV is in a split zone. Then, the follower UAV (sub-leader UAV) receives a message from the leader UAV, and the follower UAV state transitions to the hovering state (general state) before performing the mission. Subsequently, the state of the leader UAV is assigned, and the mission is performed (SPEC1).

Figures 10 and 11 illustrate the events and states of each UAV. A new leader group is created in split zone 1. UAV 2, closest to the leader UAV (UAV 1), is newly assigned as the leader and performs the mission. In split zone 2, UAV 3, closest to the leader UAV (UAV 2), is newly assigned as the leader and performs the mission. Consequently, the mission is completed by UAVs 1, 3, and 2. The transition to the general state and tributary mapping is completed. The mission completion time is 92 s, and 192 s if a single UAV is driven redundantly. Consequently, as shown in Figures 9–11, during regrouping, a new leader is assigned in each split zone, and one of the follower UAVs is assigned as a sub-leader UAV, which follows the sub-leader UAVs. Furthermore, we confirm that the control action is suitable for the designed SPEC3. The multi-UAV system tributary mapping completion time and total distance are present in Table 2. The multi-UAV completion time is 92 s, and the single UAV system is 192 s according to exploration on an overlapping path. The total distance of multi-UAV is increased compared to single UAV. Because the multi-UAV total distance is 183.72 m

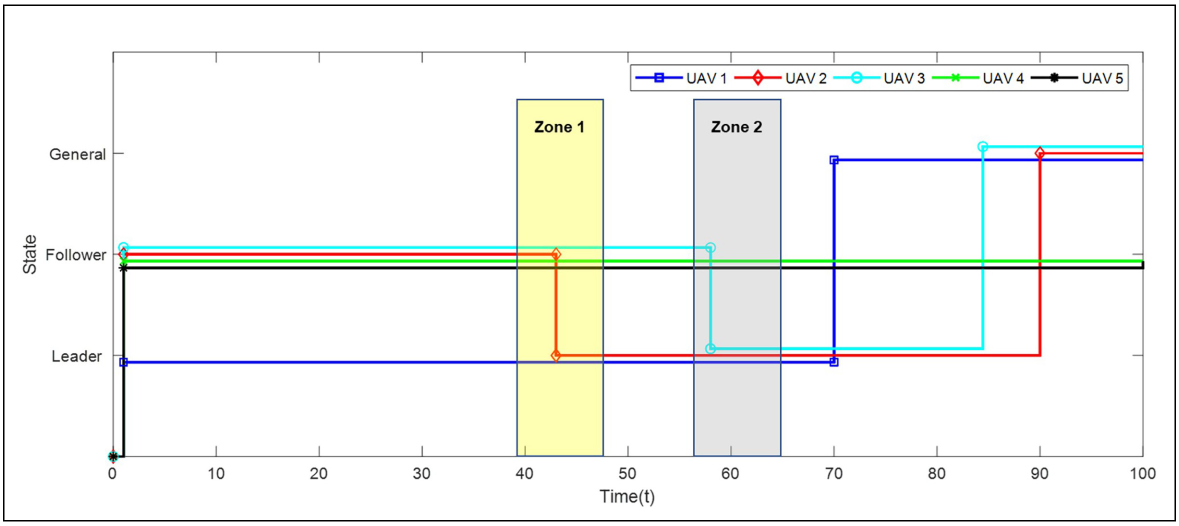

Figure 11. Experimental result of each UAV state transition.

Table 2. Experimental results of complete time and distance.

	Completion time (s)	Total distance (m)
Single UAV	192	105.92
Multi-UAV	92	183.72

UAV: unmanned aerial vehicle.

(36.16 + 40.38 + 36.04 + 37.06 + 38.08) by summing all UAV's paths. However, the path of each UAV is reduced by preventing overlapped paths.

The proposed SCT-based multi-UAV system was designed by considering controllability to solve the tributary mapping scenario, which is one of the practical applications. All events were assumed to be observable in this experiment, and the results varied depending on the controllable or observable events. If the leader UAV is lost during the tributary mapping process because of communication problems, the lost UAV path cannot be observed as an event. In this case, the newly assigned leader UAV can plan the same trajectory and perform mapping as the lost UAV. Therefore, this may not be the most efficient because it repeatedly explores the same area. This problem will be solved if we consider observability. Therefore, we assume that all events are observed, an SCT-based approach for practical applications and demonstrated its effectiveness through dynamic simulation results. The proposed SCT-based system minimized the control complexity and guaranteed scalability as the number of UAVs increases in a large-scale dynamic system.

Conclusion

In this study, we proposed an SCT-based tributary mapping approach using multi-UAVs. Each UAV was modeled as an automaton considering a split tributary environment. For the dynamic large-scale plant that was modeled, we designed a modular supervisor of each control specification for each UAV and designed control specifications based on SCT for branch mapping. The plant and specifications were modeled and designed using automata, while supervisors were synthesized using MATLAB TCT. We implemented the proposed system and verified via a physics-based simulator that satisfied the control specifications.

The formation shape was not determined, and only the leader–follower control method was presented. The multi-UAV system achieved the given goal in the simulation environment and covered all tributary regions. This method has the advantage of acquiring multiple images in a single flight. We believe this approach is suitable for practical applications using multi-UAV systems in the ecosystem and life systems. In the future, we will implement and verify tributary mapping using an SCT-based multi-UAV system via real-world experimentation.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported, in part, by the Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008473, HRD Program for Industrial Innovation) and, in part, by the Cooperative Research Program for Agriculture Science and Technology Development, Rural Development Administration, Republic of Korea, under Project PJ0147612023.

ORCID iD

Hyoung Il Son https://orcid.org/0000-0002-7249-907X

References

1. Koparan C, Koc AB, Privette CV, et al. Evaluation of a UAV-assisted autonomous water sampling. *Water* 2018; 10(5): 655.
2. Kim S, Ju C, Kim J, et al. A tracking method for the invasive Asian hornet: a brief review and experiments. *IEEE Access* 2019; 7: 176998–177008.
3. Ju C and Son HI. Investigation of an autonomous tracking system for localization of radio-tagged flying insects. *IEEE Access* 2022; 10: 4048–4062.
4. Kim B, Ju C and Son HI. Field evaluation of UAV-based tracking method for localization of small insects. *Entomol Res* 2022; 52: 135–147.
5. Seol J, Kim J and Son HI. Field evaluations of a deep learning-based intelligent spraying robot with flow control for pear orchards. *Precis Agri* 2021; 23: 712–732.
6. Jun J, Kim J, Seol J, et al. Towards an efficient tomato harvesting robot: 3D perception, manipulation, and end-effector. *IEEE Access* 2021; 9: 17631–17640.
7. Barrientos A, Colorado J, del Cerro J, et al. Aerial remote sensing in agriculture: a practical approach to area coverage and path planning for fleets of mini aerial robots. *J Field Robot* 2011; 28(5): 667–689.
8. Isikdogan F, Bovik A and Passalacqua P. RivaMap: an automated river analysis and mapping engine. *Rem Sens Environ* 2017; 202: 88–97.
9. Nuske S, Choudhury S, Jain S, et al. Autonomous exploration and motion planning for an unmanned aerial vehicle navigating rivers. *J Field Robot* 2015; 32(8): 1141–1162.
10. Scherer S, Rehder J, Achar S, et al. River mapping from a flying robot: state estimation, river detection, and obstacle mapping. *Auton Robot* 2012; 33(1): 189–214.
11. Cesare K, Skeele R, Yoo SH, et al. Multi-UAV exploration with limited communication and battery. In: *Proceedings of the 2015 IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, 26–30 May 2015, pp.2230–2235. New York: IEEE.
12. Kim J, Kim S, Ju C, et al. Unmanned aerial vehicles in agriculture: a review of perspective of platform, control, and applications. *IEEE Access* 2019; 7: 105100–105115.
13. Yao P, Xie Z and Ren P. Optimal UAV route planning for coverage search of stationary target in river. *IEEE Trans Control Syst Technol* 2017; 27(2): 822–829.
14. Ju C and Son HI. Multiple UAV systems for agricultural applications: control, implementation, and evaluation. *Electronics* 2018; 7(9): 162.
15. Yang P, Zhang A and Zhou D. Event-triggered finite-time formation control for multiple unmanned aerial vehicles with input saturation. *Int J Control Autom Syst* 2021; 19(5): 1760–1773.
16. Liu J, Fang JA, Li Z, et al. Formation control with multiple leaders via event-triggering transmission strategy. *Int J Control Autom Syst* 2019; 17(6): 1494–1506.
17. Lin C, Zhang W and Shi J. Tracking strategy of unmanned aerial vehicle for tracking moving target. *Int J Control Autom Syst* 2021; 19(6): 2183–2194.
18. Oh KK and Ahn HS. Leader-follower type distance-based formation control of a group of autonomous agents. *Int J Control Autom Syst* 2017; 15(4): 1738–1745.
19. Ju C and Son HI. A distributed swarm control for an agricultural multiple unmanned aerial vehicle system. *Proc IMechE, Part I: J Systems and Control Engineering* 2019; 233(10): 1298–1308.
20. Kim J, Ju C and Son HI. A multiplicatively weighted Voronoi-based workspace partition for heterogeneous seeding robots. *Electronics* 2020; 9(11): 1813.
21. Kim J and Son HI. A Voronoi diagram-based workspace partition for weak cooperation of multi-robot system in orchard. *IEEE Access* 2020; 8: 20676–20686.
22. Ramadge PJ and Wonham WM. Supervisory control of a class of discrete event processes. *SIAM J Control Optim* 1987; 25(1): 206–230.
23. Jiang L, Bandini F, Smith O, et al. The value of distributed high-resolution UAV-borne observations of water surface elevation for river management and hydrodynamic modeling. *Rem Sens* 2020; 12(7): 1171.
24. Jain S, Nuske S, Chambers A, et al. Autonomous river exploration. In: Mejias L, Corke P and Roberts J (eds) *Field and service robotics*. Cham: Springer, 2015, pp.93–106.
25. Gamage GW, Mann GK and Gosine RG. Discrete event systems based formation control framework to coordinate multiple nonholonomic mobile robots. In: *Proceedings of the 2009 IEEE/RSJ international conference on intelligent robots and systems*, St. Louis, MO, 10–15 October 2009, pp.4831–4836. New York: IEEE.
26. Lee G and Chwa D. Decentralized behavior-based formation control of multiple robots considering obstacle avoidance. *Intell Serv Robot* 2018; 11(1): 127–138.
27. Han Z, Guo K, Xie L, et al. Integrated relative localization and leader–follower formation control. *IEEE Trans Autom Control* 2018; 64(1): 20–34.
28. Denasi A and Misra S. Independent and leader–follower control for two magnetic micro-agents. *IEEE Robot Autom Lett* 2017; 3(1): 218–225.
29. Mwaffo V, DeLellis P and Humbert JS. Formation control of stochastic multivehicle systems. *IEEE Trans Control Syst Technol* 2021; 29: 2505–2516.
30. Unsalan C and Sirmacek B. Road network detection using probabilistic and graph theoretical methods. *IEEE Trans Geosci Rem Sens* 2012; 50(11): 4441–4453.
31. Cai K and Wonham WM. Supervisory control of discrete-event systems, 2020, https://www.control.eng.osaka-cu.ac.jp/publication/CaiWonham_20Encyclo.pdf
32. Tsalatsanis A, Yalcin A and Valavanis KP. Optimized task allocation in cooperative robot teams. In: *Proceedings of the 2009 17th Mediterranean conference on control and automation*, Thessaloniki, 24–26 June 2009, pp.270–275. New York: IEEE.
33. Lopes YK, Trenkwalder SM, Leal AB, et al. Supervisory control theory applied to swarm robotics. *Swarm Intell* 2016; 10(1): 65–97.
34. Ju C and Son HI. Modeling and control of heterogeneous field robots under partial observation. *Inform Sci* 2021; 580: 419–435.
35. Ju C and Son HI. Modeling and control of heterogeneous agricultural field robots based on Ramadge–Wonham theory. *IEEE Robot Autom Lett* 2019; 5(1): 48–55.
36. Ju C and Son HI. A hybrid systems-based hierarchical control architecture for heterogeneous field robot teams. *IEEE Trans Cybern* 2021; 53: 1802–1815.
37. Wonham WM. *Supervisory control of discrete-event systems*. Cham: Springer, 2015.