

A Multi-UAV System for Exploration and Target Finding in Cluttered and GPS-Denied Environments

Xiaolong Zhu^{1,2}, Fernando Vanegas¹, Felipe Gonzalez¹, Conrad Sanderson^{2,3}

¹ Queensland University of Technology, Australia

² Data61 / CSIRO, Australia

³ Griffith University, Australia

Abstract—The use of multi-rotor Unmanned Aerial Vehicles (UAVs) for search and rescue as well as remote sensing is rapidly increasing. Multi-rotor UAVs, however, have limited endurance. The range of UAV applications can be widened if teams of multiple UAVs are used. We propose a framework for a team of UAVs to cooperatively explore and find a target in complex GPS-denied environments with obstacles. The team of UAVs autonomously navigates, explores, detects, and finds the target in a cluttered environment with a known map. Examples of such environments include indoor scenarios, urban or natural canyons, caves, and tunnels, where the GPS signal is limited or blocked. The framework is based on a probabilistic decentralised Partially Observable Markov Decision Process which accounts for the uncertainties in sensing and the environment. The team can cooperate efficiently, with each UAV sharing only limited processed observations and their locations during the mission. The system is simulated using the Robotic Operating System and Gazebo. Performance of the system with an increasing number of UAVs in several indoor scenarios with obstacles is tested. Results indicate that the proposed multi-UAV system has improvements in terms of time-cost, the proportion of search area surveyed, as well as successful rates for search and rescue missions.

I. INTRODUCTION

The use of multiple Unmanned Aerial Vehicles (UAVs) has recently gained more attention for tasks such as real-time monitoring, Search and Rescue (SAR), remote sensing, delivery of goods, precision agriculture, and infrastructure inspection [1]–[5]. In SAR settings, UAVs have been considered due to their manoeuvring capabilities (flying at various speeds, hovering, agile access to narrow spaces), making them suitable for missions that can be dangerous or difficult for human beings [6]–[10].

Research work focusing on multiple UAVs for SAR missions can be broadly divided into two main areas: (i) swarm systems, which consist of a large number of smaller drones with lower quality sensors and a lower level of autonomy [11]–[13], and (ii) cooperative systems, which contain only several larger UAV platforms with more sensors and more powerful on-board processors to enable more complex on-board decisions than smaller drones [2], [14], [15]. There is considerable amount of research on SAR missions in GPS-denied environments with a single UAV [16]–[20]. In contrast, using a team of UAVs to find a target in cluttered and GPS-denied scenarios with efficiency and cooperation is still a challenging problem.

Most of the existing research using multiple UAVs and Markov Decision models follows two approaches:

- Partially Observable Markov Decision Processes (POMDP). Using a real-time algorithm to solve an optimal or approximate policy of an agent or the whole team. This approach creates a universal framework, which can be implemented in a variety of environments, including dynamic and partly known environments [21], [22].
- Reinforcement Learning (RL), including Deep RL. The best policy is obtained for a given scenario and time cost with high performance hardware to accomplish the training process [17], [23]. One drawback is that any change in the formulation or model can lead to a more training and fine-tuning processes which are time-consuming.

In this paper, we extend our previous work for searching to find a target without the use of GPS. In [24], we proposed a decentralised decision-making system based on POMDPs containing an information sharing module for target finding by multiple UAVs. That system was assigned several possible target positions within a known map, GPS signal for accurate localisation was available, and was assisted by the information sharing module to achieve an efficient and scalable teamwork. Here, we exploit the use of a message exchange module in order to apply the same approach to search missions without known target position. For this purpose, the search environment is divided into several grids, and the mission is divided into several tasks to explore the grids to find the target. The feedback of tasks is exchanged among the UAVs through the communication channel. This allows for a decentralised and scalable system which is able to cooperatively perform in SAR applications with an affordable computational load and complexity for the on-board processors.

Furthermore, we propose a method to assist UAV localisation in the GPS-denied environments, which fuses the distance to an obstacle from six close range laser sensors with the known map of the environment. These laser sensors help each UAV in the system to avoid collisions with obstacles and other UAVs.

We continue the paper as follows. Section II introduces POMDPs for single and multiple agents system; Section III describes the problem addressed in this work; Section IV details the UAV system architecture; Section V proposes a decentralised POMDP formulation to address the problem; Section VI details the experiment setup; Section VII presents the results; Section VIII summarises the main findings and lists possible avenues for future work.

II. BACKGROUND

In this work, UAVs are treated as autonomous agents required to accomplish a goal by a set of predetermined actions. Decision making, which can be ranged from the classical utility maximisation (quantitative) by presented complete and static information to a complex, dynamic, and probability planning, is the essential functionality of these agents [25]. In this work, we consider a decision making framework for these agents.

A. Single Agent MDPs and POMDPs

In the classical quantitative decision region of single agent, Markov Decision Processes (MDPs) have been widely studied as an effective mathematical framework for sequential decision making in stochastic domain [26]. The objective of MDPs is to compute a strategy or a policy. This policy acts a function that enables the agent to choose given actions at a particular decision point depending on the agent's state. The policy assigns specific values to actions which depend on the reward function. The reward function represents the goals, objectives or penalties for the agent for arriving at a particular state. An optimal policy maximise the utility of the agent for all possible sequences of actions and states. However, MDPs are based on a critical assumption that the agent is able to obtain a complete observation at each decision point. However, this assumption is hardly achievable with the variety of limitations of robotics and sensors in real scenarios. Thus, it is desirable to extend MDPs to handle the incomplete observational capabilities of agents and to consider the inaccurate or insufficient observations in the model.

This extension of MDPs is well-known as Partially Observable Markov Decision Processes (POMDP). POMDP incorporates the partial observability of the agent and the uncertainties from sensors and environments into the stochastic items. Formally, a POMDP is a tuple described as $\langle S, A, T, \Omega, O, R, \gamma \rangle$ [26], where:

- S is a (finite) set of all the environment states for the agent;
- A is the set of all possible actions of the agent;
- $T(s, a, s')$ is the probability of agent in state $s' \in S$ after performs an action $a \in A$ at previous state $s \in S$, which equals with $P_r(s' | s, a)$;
- Ω is the set of all possible observations;
- $O(s', a, o)$ is the probability of observing $o \in \Omega$ at state s' after performed the action a ;
- R is the set of rewards for each chosen action a at every state s ;
- $\gamma \in [0, 1)$ is the discount factor.

The actual state of the system cannot be directly observable from the tuple. Instead, a state called the belief-state $b(s)$ is obtained by using the Bayes rule, which is a probability distribution of the agent at all the possible states in its state-space at a certain time. Due to the Markov assumption, it can be updated with a Bayesian filter for every

action–observation pair:

$$b'(s') = \eta \Omega(s', a, o) \sum_{s \in S} T(s, a, s') b(s) \quad (1)$$

where η acts as a normalising constant such that b' remains a probability distribution.

The objective of a POMDP is to find a policy that maps beliefs into actions in the form $\pi(b) \rightarrow a$, so that the total expected reward is maximised. This expected reward gathered by following π starting from belief b is called the value function:

$$V(b) = E \left[\sum_{t=0}^h \gamma^t \sum_{s \in S} R(s, \pi(b_t)) b_t(s) | b_0 = b \right] \quad (2)$$

Therefore, the optimal policy π^* is the one that maximises that value function: $\pi^*(b) = \arg \max_a V(b)$.

B. Multiple Agents POMDPs

In the case of multiple agents in sequential decision making under uncertainty, an extension from single agent POMDP framework for multi-agent POMDP is needed. The generalisation of the POMDP to multiple agents can be used to model a team of cooperative agents that are situated in a partially stochastic and partially observable environment. Formally, a multi-agent POMDP can be defined as a tuple $\langle I, S, \mathbb{A}, T, \mathbb{O}, O, R, \gamma, h \rangle$ [27], where:

- I is a finite set of n agents, $i \in I$;
- S is a (finite) set of all the environment states for the agents;
- \mathbb{A} is the set of joint actions: $\mathbb{A} = \times_{i \in I} A_i$, A_i is the set of available actions of i -th agent, which can be unique for each agent;
- T is a set of state transition probabilities: $T = S \times \mathbb{A} \times S \rightarrow [0, 1]$, the probability of transitioning from previous state $s \in S$ to current state $s' \in S$ when set of joint actions $\vec{a} \in \mathbb{A}$ are taken by each agent. Hence, $T(s, \vec{a}, s') = P_r(s' | s, \vec{a})$;
- \mathbb{O} is the set of joint observations: $\mathbb{O} = \times_{i \in I} \Omega_i$, Ω_i is a set of observations available to i -th agent;
- O is the observation probability function: $T = \mathbb{O} \times \mathbb{A} \times S \rightarrow [0, 1]$, the probability of seeing the set of joint observations $\vec{o} \in \mathbb{O}$ given the set of joint actions $\vec{a} \in \mathbb{A}$ were taken which results in state $s' \in S$, Hence, $O(\vec{o}, \vec{a}, s') = P_r(\vec{o} | s', \vec{a})$;
- R is the reward function: $R = S \times \mathbb{A} \rightarrow \mathbb{R}$, the immediate reward maps state $s \in S$ and performing the set of joint actions $\vec{a} \in \mathbb{A}$ and is used to specify the goal of the agents;
- $\gamma \in [0, 1)$ is the discount factor for the reward function R ;
- h is the number of steps after current state, which the problem terminates.

In a Multiple Agent System (MAS) problem, the goal is to calculate an optimal joint policy $\vec{\pi}^* = \{\pi_1, \dots, \pi_n\}$ that maximises the accumulated discounted reward via the reward function (R) as in process in the POMDP model.

Two approaches are commonly used: a centralised Multi-agent POMDP (MPOMDP) model and a Decentralised POMDP (Dec-POMDP) model the formulation.

1) *Multi-agent POMDP*: In the MPOMDP case, there is a central node or agent, which has communication and access to obtain the complete current states \vec{s} and joint observations \vec{o} of all other agents, to compute an optimal global policy $\vec{\pi}^*$. This approach is able to maximise the cooperation and coordination as a MAS. It means that the centralised system ensures the efficiency and collaboration in a mission. On the other hand, the system requires and relies on a stable network connect on, noise-free & instant communication, and large bandwidth for sufficient data exchange between agents [28]. Moreover, this approach is difficult to apply in complex and dynamic environments in a limited reaction time. These demanding constraints narrow down the application domain of a MPOMDP system.

2) *Decentralised POMDP*: A Dec-POMDP model is versatile for several applications when compared to the centralised one. Each agent in the decentralised system locally plans its own decision mainly based on its local observations Ω_i and actions A_i . The Dec-POMDP model has more flexibility of the various grades of communication, from none to perfect and between indirect and direct [29].

3) *Complexity of the MAS*: Despite all of the benefits of a MAS, the computational complexity of the system has prevented its widespread applications in the real world. In the past decade, multiple research has been conducted to reduce the expensive consumption of Dec-POMDP both on advanced optimal policy algorithms and specific used models [25], [27], [29]. However, the structure of the MPOMDP and Dec-POMDP models is not altered and the computational complexity of the problem exponentially increases with the number of agents [30]. Bernstein *et al.* [30], [31] have shown that solving either optimally or approximately a Dec-POMDP remains NEXP-complete. This intractable issue renders the application of the MPOMDP and Dec-POMDP models to simplest cases and limited for real word applications.

III. PROBLEM DESCRIPTION

Using multiple UAVs for SAR missions requires the system to be capable of dealing with five main issues:

a) *Search Area Coverage*: In any SAR mission, the search area coverage is an essential part. A high explored coverage rate means the designed system is more likely to find the target, such as a victim or trapped person.

b) *Cost Time*: Time is the crucial metric within the SAR context. In fire disasters a few minutes can make the difference between finding a person dead or alive.

c) *Unavailable GPS Signal*: Use of GPS signals is the main approach for determining accurate positions of autonomous UAVs. However, GPS signals can be limited or unavailable in urban canyons and tunnels. There are also obstacles which can obstruct GPS signals.

d) *Unstable Communication Connection*: The system needs to be robust to an unstable communication network, due to interference or natural signal blockage.

e) *Collision Free*: Challenges in cluttered space determines size UAVs smaller than 30 cm in diameter to be useful.

This paper focuses on a scalable, decentralised multi-UAV system for navigation, exploration and target finding in a GPS-denied and cluttered environment under uncertainty. The SAR scenarios in this paper are based on an indoor area which contains pillars and lounges. The UAVs are deployed from an empty space on the other side of entry of the hall, shown in Fig. 2. UAVs need to navigate, search and find a unknown target represented by an augmented reality (AR) tag attached on the ground, via using a downward-facing camera. A map of the inner space is known to the multi-UAV system. The mission is to find the target and send its position through the network. In the mission, an unstable and noisy communication network (e.g. WiFi or radio) is also considered. Furthermore, avoidance of collisions with obstacles and other UAVs is also required. It is assumed that there are no changes in flight conditions and there is no wind that will cause a disturbance of UAV flight performance. Ground level is at sea level. It is also assumed that each UAV only has access to the computational capacity of its own on-board processor.

IV. SYSTEM ARCHITECTURE

In order to maintain the vital advantages of a decentralised system, such as scalability and reliability, the whole multi-UAV system consists of several identical single-UAV systems which have a design taking into account the concepts of modularity and versatility, as shown in Fig. 1. Each UAV system contains six modules grouped into four subsystems: Mission Planning, Positioning, Observation, and Information Exchange. The Mission Planning subsystem includes the modules for motion planner, motion execution and data processing. The Positioning subsystem can have various combinations depending on the application. In this paper, the Positioning subsystem consists of an inertial odometry module and a range sensor module. The Observation subsystem contains a target detection module and an observation processing module. The Information Exchange subsystem consists of a communication network and an observation sharing module.

A. Motion Planner Module

The motion planner module is one key part of the multi-UAV system. Its primary goal is to solve the optimal policy (π^*) and to select an action (A_i) in order to accomplish the mission. A detailed description of the decentralised POMDP used is presented in Section V. In this paper, the motion planner solver is based on the TAPIR open source software [32], which is a fast online solver for large and continuous state space POMDP by using the Adaptive Belief Tree (ABT) algorithm [33]. TAPIR was formulated as a motion planner solver, using a C++ implementation. (<https://github.com/rdl-algorithm/tapir>)

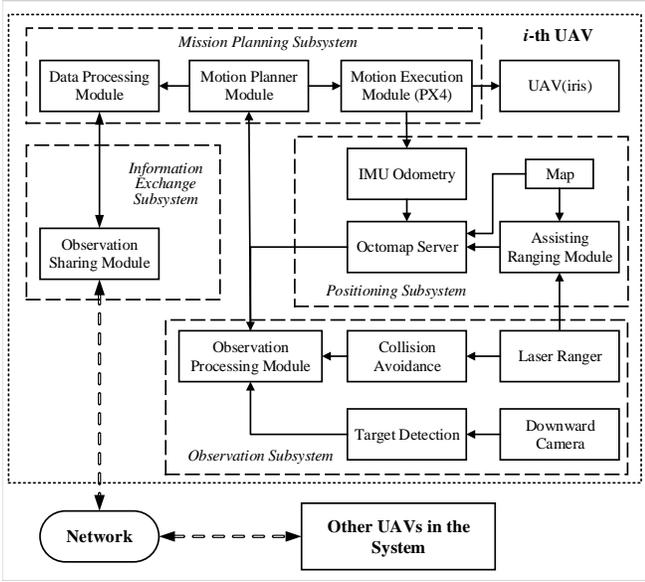


Fig. 1. Multi-UAV system architecture.

B. Motion Execution Module

After a set of actions is defined in the system formulation, the chosen action (A_i) by the motion planner solver needs an autonomous UAV flight control system for manoeuvring. PX4, an open source software developed by the Dronecode Project [34], is widely used as the flight control unit (FCU) in several UAV applications [20], [35]. The flight stack layer of the PX4 architecture contains a pipeline of feedback loop based flight controllers for various types of UAVs, such as multi-rotors, fixed-wing and VTOL, and position & altitude estimators. A high precision inertial measurement unit (IMU) is contained in PX4 as the inertial odometry. The PX4 also has a middleware layer, which contains the device drivers for onboard sensors, communication interfaces, and a simulation layer.

C. Data Processing Module

In this paper the map of the environment is known and divided into grids of the same size. This allows each UAV to explore a small area, which covers a grid, to find the target. With the help of this module, the UAV is able to send an observation information to other UAVs in the team in order to improve the efficiency of the whole system. There are two main functions of this module:

- Searching and locating a new empty grid in the map by accessing the UAV current target particles in the system belief;
- Recording the coordinates of this grid and sending it to the observation sharing module.

D. Octomap Server with Map

One of the main parts of the positioning subsystem is the Octomap server which uses the 3D occupancy voxels to represent the obstacles and boundaries within the flying area. The map of the environment contains walls, doors

and primary obstacles. A Gazebo simulation world is built based on the given map, and an equivalent occupancy map is also created for the Octomap server as the probability environment for the POMDP solver. The IMU odometry from the PX4 provides the basic estimate position of the UAV, the Octomap server then sends position estimate data to the motion planner module and the observation processing module in the occupancy map.

E. Assisting Ranging Module

The primary purpose of this module is to calibrate the estimated position provided from the inertial odometry within the flight. There is a multi-ranger sensor that uses 5 laser sensors to measure the distance in the directions front, back, left, right, and up of the UAV. This multi-ranger sensor is a relatively precise measurement in a close range. With this sensor, the UAV gets a precise localisation to the motion planner when it nears an obstacle. This module is also used for a collision avoidance when the safety zone of a UAV is occupied by an obstacle. This reaction is priority than a computed action of the motion planner.

F. Observation Processing Module

The observation processing module is designed to sense the objectives, obstacles and target. The multi-ranger sensor is used for identifying obstacles in the flying area in order to avoid a collision between the UAV and the obstacles. In order to avoid UAV-to-UAV collision, this module always monitors other UAVs flight altitudes shared by the communication network (in this paper, WiFi is used as the connection between the UAVs). The module generates an observational input to the motion planner to avoid the UAV within same altitude with its peers. The downward-facing monocular camera is used for recognising the ground target via an AR tag on the top of the target. This allows the system to calculate the position and orientation of the target in the global frame based on the UAV pose and the AR tag position in the image frame. Finally, the observation processor collects data from various sensors and combines all the shared observation in the network with its observation to update the motion planner.

G. Observation Sharing Module

This module is the critical part of the whole system. With an effective observation sharing module, the multi-UAV system performs to cooperatively accomplish the mission, instead of each UAV working independently. The module obtains the data from the solver and simplifies the local observations to a message and frequently publishes the message through the network. A typical message contains three parts of observation: target detection status & localisation, unified ID of the latest explored grid with this UAV ID, and all explored grids' ID. This kind of structure of the published message is convenient for monitoring and updating the observation.

V. PROBLEM FORMULATION

As described in Section II, due to the complexity of the formal Dec-POMDP and its many extensions, it is unaffordable for a onboard computer to solve the entire problem, even using state-of-the-art Dec-POMDP solving algorithms. In this paper, an observation-oriented Dec-POMDP (Ob-Dec-POMDP) model was therefore extended and modified and used to formulate and solve the problem described in Section III. This Ob-Dec-POMDP model is expanded from our previous work [24], which maintains the scalability and cooperation as a MAS without an intractable computational load for the current onboard processor, as a compromise and combination between the theoretical mathematic model and the real-world implementation. The primary purpose of this Ob-Dec-POMDP formulation is to accomplish a comprehensive mission as an efficient cooperative MAS within fast reaction time in cluttered environments. The Ob-Dec-POMDP can be described as a tuple $\langle I, S, A, T, \{\Omega_l, \Omega_g\}, O, R, \gamma, h \rangle$.

A. State Variables (S)

As I represents the finite set of UAVs and can be used as a identification in the multi-UAV system, the state of each UAV is described as a tuple $S = \langle i, P_r, P_t, V_m, d_c, t_f \rangle$, where i is the identification of this UAV, $P_r = \{x_r, y_r, z_r, \psi_r\}$ are the position and orientation of this UAV based on the world coordinates, $P_t = \{x_t, y_t, z_t, \psi_t\}$ are the position and orientation of the target in the world coordinates, $V_m = \{v_x, v_y, v_z, v_\psi\}$ are the flight velocities of UAV on longitudinal, lateral, vertical, and yaw axes on the UAV body frame, then $d_c = \{d_{xc}, d_{yc}, d_{zc}\}$ are the safety distance from obstacles on the UAV local frame, and the last part t_f is the flight manoeuvre time for each UAV.

B. Actions (A)

The set of actions (A) in this formulation contains several actions: Forward, Backward, Go-left, Go-right, Go-up, Go-down, Turn-left, Turn-right and Hovering, where:

- Forward: flight forward with a distance $v_x \cdot t_f$ on the UAV local coordinates;
- Backward: flight backward $-v_x \cdot t_f/2$ to avoid a front obstacle;
- Go-left & Go-right: flight to left or right $v_y \cdot t_f$ from a lateral approaching UAV or obstacles;
- Go-up & Go-down: changing the altitude to avoid a obstacle or other UAVs and for a different field of view (FOV) of the downward-facing camera;
- Turn-left & Turn-right: change the yaw angle of the UAV by $v_\psi \cdot t_f$;
- Hovering: holding the UAV current pose for more detailed observation.

In this paper, the actions (A) are divided into two categories based on its purpose: the first one is the search actions (Forward, Turn-left, Turn-right, Hovering, Go-up and Go-down) which are performed the UAV to explore the scenarios to find the target; the second is the avoidance actions (Backward, Go-left, Go-right, Go-up and Go-down) that are

only activated on the collision avoidance situation, when the multi-ranger sensors detects an object or another UAV in the UAV safety zone.

C. Observations ($\{\Omega_l, \Omega_g\}$)

In order to limit the computational cost while keeping the scalability of the system, an extended observation approach is designed and tested in real-world scenarios based simulations. The observations Ω consists of two parts:

a) *Local Observation* (Ω_l): this is obtained through the local on-board sensors of each UAV, which contains the status of the target detection under uncertainty, the position of the detected target in the FOV of the downward-facing camera, the ID of the explored grid, the status of avoidance mode (keeping the safety zone of UAV), the distance from the surrounding obstacles in the map.

b) *Global Observation* (Ω_g): the global observation from other UAVs are provided as the a latest observation shared message only contained the useful and simplified observation information via the system communication. A shared global observation message consists of: the status of the target detection, the global position of the target in the map with the detected UAV ID, the latest explored grid ID with the ID of the searching UAV, and the IDs of all of explored grids.

D. Transition Function (T)

The transition function (T) models the UAV dynamic response to motion commands that are represented as the step inputs. In order to represent the i -th UAV dynamics, a kinematic model is described as follows:

$$P_r(t+1) = P_r(t) + T_r(t) \Delta P_r(t) \quad (3)$$

$$\Delta P_r(t) = V_m(t) \Delta t + P_{wt}(t) \quad (4)$$

where

$$T_r(t) = \begin{bmatrix} \cos(\psi_{rt} + \psi_{wrt}) & -\sin(\psi_{rt} + \psi_{wrt}) & 0 & 0 \\ \sin(\psi_{rt} + \psi_{wrt}) & \cos(\psi_{rt} + \psi_{wrt}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In the above equations, $T_r(t)$ is the dynamic change matrix at time step t , $P_r(t)$ is the position of i -th UAV in the world frame at time step t and $P_{wt}(t) = \{x_{wt}(t), y_{wt}(t), 0, \psi_{wt}(t)\}$ is the pose uncertainty during the flight. The position of each UAV is determined by calculating the change in position due to its current velocity (Eq. 3), which is controlled by the motion planner module and is selected according to the commanded action. A transformation from the UAV's frame to the world frame is also calculated in these equations.

The PX4 controllers use a multi-copter control architecture, which is a standard cascaded control architecture with mixed of P and PID controllers [34]. The PX4 controllers are capable of providing a comprehensive pose output for the UAV state (S_t) on the world frame with a solved action input.

E. Reward Function (R)

Table I lists two possible options for the reward function (R): Option A sets equal cost for all the search actions, and Option B is designed for a more aggressive exploration e.g. less hover. As well as a positive reward when each UAV explores a new grid. The reward function at each step consists of: (i) the reward of finding the target and explored a new grid in the map, (ii) the penalty of collision with an obstacle or flight out of operational space, and (iii) the cost of each manoeuvre.

F. Uncertainty

The three main sources of uncertainty of the system consist of the localisation uncertainty of the airborne UAVs, the position uncertainty of the target, and the shared observations uncertainty in the communication.

1) *Localisation of UAVs*: The localisation uncertainty for each UAV in this formulation covers two parts. In the first part, the uncertainty in localisation of system once it is initialised (on the $x-y$ plane) is $P_{wd} = \{x_{wd}, y_{wd}, 0, \psi_{wd}\}$, in which the UAVs will hover for 5 seconds to initialise sensors and network connection after take-off, represented by a Gaussian probability distribution ($\mu = 0.0m$, $\sigma = 0.5m$). In the second part, the UAVs' pose uncertainty during the flight is $P_{wt} = \{x_{wt}, y_{wt}, 0, \psi_{wt}\}$, which is described by the Gaussian distribution ($\mu = 0.0m$, $\sigma = 0.2m$) & ($\mu = 0.0^\circ$, $\sigma = 5.0^\circ$). We use a downward laser sensor to calibrate the altitude and assume all the altitude data is accurate for the system through the flight. These localisation uncertainty values are updated through the transition function.

2) *Target position*: The position uncertainty of the target $P_t = \{x_t, y_t\}$ is represented by a uniform distribution in the scenario.

3) *Shared message*: The uncertainty in shared observations is caused by noisy communications. This type of noise is modelled with a confidence factor $\lambda = 0.9$, which means 90% of particles representing the target positions are removed from the current belief based on the shared message of new explored grid by another UAV. Moreover, there is a double-check mechanism used for two latest shared messages to confirm the data in the messages is correct and true. These observation-related uncertainties are updated by the observation sharing module.

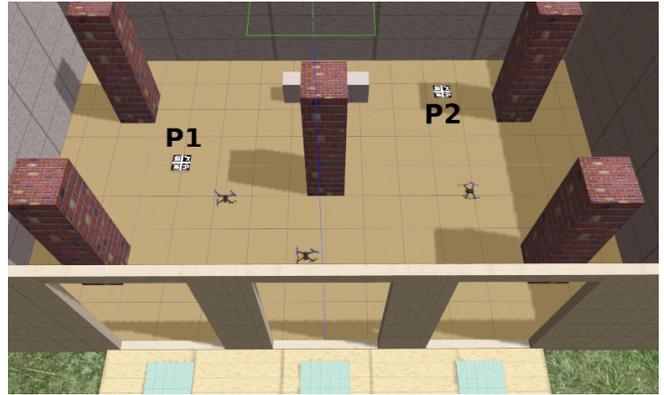
VI. EXPERIMENTS

In order to test and evaluate the performance of the proposed framework, two sets of comparative experiments have been performed. The simulation environment consists of Robotic Operating System (ROS), Gazebo, Octomap, Rviz as visualisation tool, and PX4 at software in the loop (SITL) mode as UAV flight control unit. An off-the-shelf UAV model (3DR Iris) is used as the aerial platform in the simulations. The mission of the multi-UAV system is to explore an indoor space to find a target while avoiding obstacles in two types of complexity scenarios, as shown in Fig. 2. The parameters of the system were set as $t_f = 1.0$ sec, $v_x = 0.8$ m/s, $v_y = 0.4$ m/s,

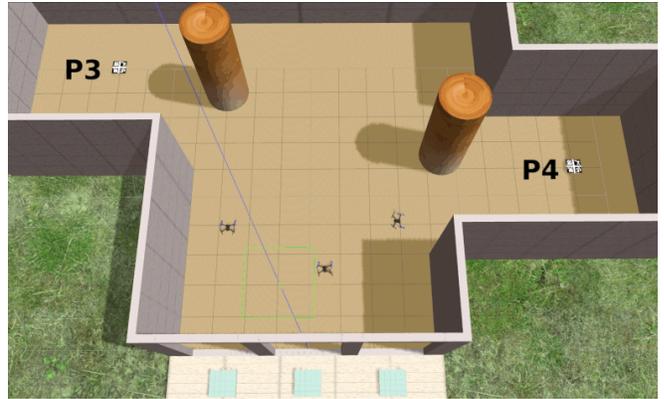
TABLE I
REWARD FUNCTION SETS

State Status & Actions	Reward or Cost Value	
	Option A	Option B
Detecting the target	300	300
Explored a new grid	60	60
Hitting an obstacle	-700	-350
Out of flying zone	-600	-600
Forward	-10	-3
Turn-left & Turn-right	-10	-5
Go-up & Go-down	-10	-10
Hover	-10	-50
Backward, Go-left & Go-right*	400	400
Go-up & Go-down*	400	400

*only in the avoidance actions



(a) Scenario one, with target locations P1 and P2



(b) Scenario two, with target locations P3 and P4

Fig. 2. Simulation environments with two unique scenarios (roof not shown for illustration purposes).

$z_t = 0.5$ m/s, $v_\psi = 30^\circ/s$, $d_c = \{0.3m, 0.3m, 0.3m\}$. In all the experiments involving scenario one, the UAVs took off from the same locations $(-3.5, -1.0, 0.2)$, $(0.0, -1.0, 0.2)$ and $(3.5, -1.0, 0.2)$ to find the target located in either $P1 = (-4.0, 5.0, 0.0)$ or $P2 = (3.5, 7.5, 0.0)$. While in scenario two, the UAVs took off from the same locations $(-2.5, -1.0, 0.2)$, $(0.0, -1.0, 0.2)$ and $(2.5, -1.0, 0.2)$ to find the target located in either $P3 = (-7.0, 10.0, 0.0)$ or $P4 = (9.0, 6.0, 0.0)$. If the UAVs system can not find the target within a maximum flight time of $t_{max} = 600$ seconds, the system reports a failure to find the target.

TABLE II

TARGET DETECTION RATE OF THE THREE STRATEGIES IN THE SECOND EXPERIMENT.

Target Location	Success Rate		
	Independent	Divided	Informed
P1	90%	97.5%	100%
P2	67.5%	87.5%	95%
P3	37.5%	67.5%	82.5%
P4	42.5%	72.5%	87.5%

A. Experiment One

In the first experiment, two-UAV and three-UAV simulations were implemented to search and find the target in scenario one, in order to compare the reward settings in Table I, under the performance metric of *time to find the target* (search time). For each multi-UAV system, 40 simulations were performed, with same initial positions of UAVs and two separate target positions.

B. Experiment Two

The second experiment compares three strategies of three-UAV simulations in the same test zones (scenario one and two):

- Independent: this strategy sets all three UAVs as separate independent agents, where each UAV attempts to accomplish the mission by itself;
- Divided: in this strategy, the entire search area was divided into three equal zones, and each UAV attempts to search for the target in a assigned zone without knowing any information about the other co-workers;
- Informed: the three UAVs work jointly, and exchange messages as described in the Section IV.

All the above strategies were evaluated on 40 runs, each with the same initial positions of UAVs and two separate target locations in each scenario. In this comparison, Reward Option B settings were used for all strategies to compare the search time and success rate of finding the target.

VII. RESULTS

Fig. 3 shows the results for the two reward settings (as elucidated in Section V-E). Overall, it is clear that the three-UAV system spent less time to find the target than the two-UAV system with the same combination of reward options and target locations. The reduction in median target finding time was about 23% when one UAV is added to the system. We conjecture that the performance of a multi-UAV system may be further enhanced by increasing the number of UAVs. However, the performance gains for a large group are likely to be marginal, while collision risk is likely to increase.

The results in Fig. 3 also show that a tailored reward setting can enhance performance. Specifically, reward option B provided a range of performance increase from 13.9% to 37.4% when compared to option A. Due to the unequal value of cost in option B of the reward function, the motion planner prefers the less costly action (Forward) rather than the more

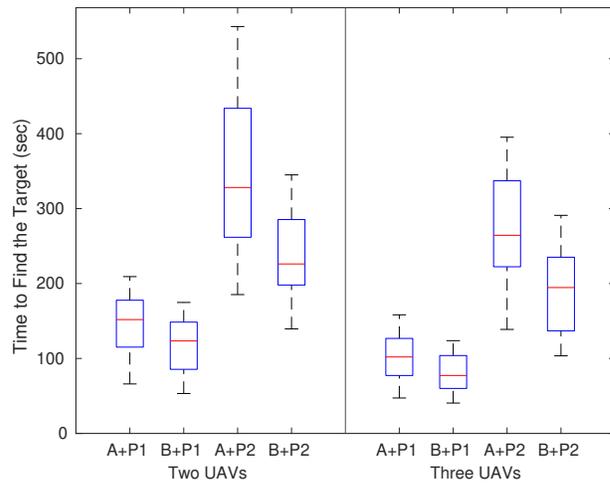
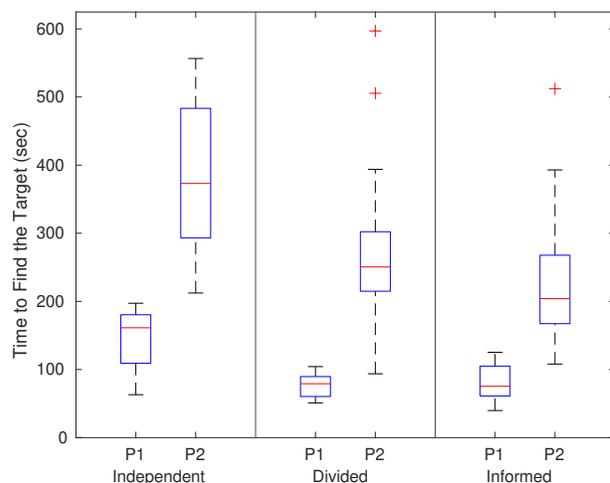
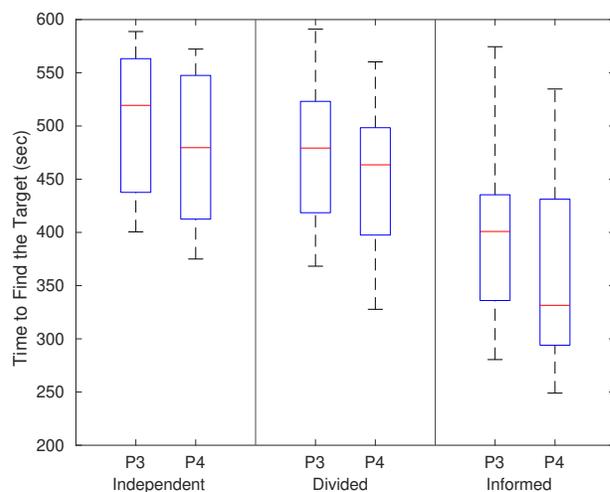


Fig. 3. Target finding time with various combinations of Reward options (A and B) and target locations (P1 and P2).



(a) Scenario one with target locations P1 and P2



(b) Scenario two with target locations P3 and P4

Fig. 4. Target finding time for three unique strategies for four target locations (P1, P2, P3 and P4).

costly action (Hover). This type of decision helped the UAVs to explore more area and reduce the search time. The target location was a critical factor that affected performance. Comparing the target locations (P1 and P2), the median time cost increased from P1 to P2, and the interquartile range (IQR) also increased noticeably between these two locations with same reward option. This is expected, as target P2 is located in a far corner and is hence a more demanding task than target P1.

Fig. 4 shows the effect of the three strategies on target finding time. Overall, the time taken was reduced from Independent to Informed. The IQR of the results in the Independent setting were significantly larger in contrast with the Divided and Informed settings. The Independent strategy performed poorly and its performance widely fluctuated compared to the other two categories (Divided and Informed). For the Informed strategy, the time to find target P1 was slightly longer than for Divided. We conjecture this may be due to the target being located at the border of two zones, where it can be found by two out of three UAVs. For the P2 target, the time taken in the Informed setting was lower than the Divided setting. The results for P3 and P4 indicate that the Informed strategy has a significant advantage over the Independent and Divided strategies for the more complex scenario.

Table II contrasts the target detection rates for all strategies. For all target locations, the Independent strategy was considerably worse compared to the Divided and Informed strategies, especially for demanding places (P3 and P4) in an intricate area. However, the Informed setting kept a consistent high successful rate in all simulations, while the rate of the Divided generally dropped from 97.5% of P1 to 72.5% of P4. Overall, the Informed strategy obtained the highest success rates.

VIII. CONCLUSIONS AND FUTURE WORK

This paper presented an approach and framework for multi-UAV system search to find a target in an indoor environment without GPS/GNSS. The framework is able to consider the uncertainties present in the formulation, by using the proposed Ob-Dec-POMDP as the underlying model, in order to control computational cost without compromising the benefits stemming from a multi-agent system. The extended observation and the observation exchange subsystems are the critical and novel feature to keep the scalability and cooperation of the whole system, which can be implemented with on-board processors.

A more specific and goal-oriented reward setting provides advantages in the decision process in comparison to an equal action cost setting. In order to navigate and localise UAVs in a GPS-denied environment, the proposed system uses a framework which combines a close range detector and a baseline map. A contrast test of various strategies is simulated to demonstrate the influence on system performance with several cooperation strategies.

The main contribution of the paper is the demonstration of the presented framework which can be implemented for

a MAS and target search in a GPS-denied environment. The information sharing strategy balances the computational cost while maintaining the system efficient and scalable.

Further avenues of research include finding and merging an affordable Simultaneous Localisation and Mapping (SLAM) system into the multi-UAV system framework. Moreover, real-world experiments which explore further constraints on the communication network systems can be performed to illustrate versatility of the designed framework.

ACKNOWLEDGEMENTS

We thank Dr. Hanna Kurniawati for the open source software TAPIR that was used as the online POMDP solver.

REFERENCES

- [1] J. Joaquin, B. C. Arrue, and A. Ollero, "Cooperative search, reconnaissance, surveillance," in *Encyclopedia of Aerospace Engineering*. John Wiley & Sons, Jun 2016.
- [2] Z. Kashino, G. Nejat, and B. Benhabib, "Multi-UAV based autonomous wilderness search and rescue using target iso-probability curves," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, pp. 636–643.
- [3] C. Ju and H. Son, "Multiple UAV Systems for Agricultural Applications: Control, Implementation, and Evaluation," *Electronics*, vol. 7, no. 9, p. 162, Aug 2018.
- [4] R. Shakeri, M. A. Al-Garadi, A. Badawy, A. Mohamed, T. Khattab, A. K. Al-Ali, K. A. Harras, and M. Guizani, "Design Challenges of Multi-UAV Systems in Cyber-Physical Applications: A Comprehensive Survey and Future Directions," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3340–3385, oct 2019.
- [5] H. Shakhatareh, A. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned Aerial Vehicles: A Survey on Civil Applications and Key Research Challenges," *arXiv:1805.00881*, pp. 1–58, Apr 2018.
- [6] H. A. Tair, T. Taha, M. Al-Qutayri, and J. Dias, "Decentralized multi-agent POMDPs framework for humans-robots teamwork coordination in search and rescue," in *Int. Conf. Information and Communication Technology Research (ICTRC)*, May 2015, pp. 210–213.
- [7] J. Scherer, S. Yahyanejad, S. Hayat, E. Yanmaz, T. Andre, A. Khan, V. Vukadinovic, C. Bettstetter, H. Hellwagner, and B. Rinner, "An autonomous multi-UAV system for search and rescue," in *First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*. ACM, 2015, pp. 33–38.
- [8] C. T. Recchiuto and A. Sgorbissa, "Post-disaster assessment with unmanned aerial vehicles: A survey on practical implementations and research approaches," *Journal of Field Robotics*, vol. 35, no. 4, pp. 459–490, 2018.
- [9] M. Liu, K. Sivakumar, S. Omidshafiei, C. Amato, and J. P. How, "Learning for multi-robot cooperation in partially observable stochastic environments with macro-actions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, sep 2017, pp. 1853–1860.
- [10] R. Z. B. Bravo, A. Leiras, and F. L. Cyrino Oliveira, "The Use of UAVs in Humanitarian Relief: An Application of POMDP-Based Methodology for Finding Victims," *Production and Operations Management*, vol. 28, no. 2, pp. 421–440, feb 2019.
- [11] S. Perez-Carabaza, E. Besada-Portas, J. A. Lopez-Orozco, and J. M. de la Cruz, "Ant colony optimization for multi-UAV minimum time search in uncertain domains," *Applied Soft Computing*, vol. 62, pp. 789–806, Jan 2018.
- [12] Y. Zhao, X. Wang, C. Wang, Y. Cong, and L. Shen, "Systemic design of distributed multi-UAV cooperative decision-making for multi-target tracking," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 1–2, pp. 132–158, Mar 2019.
- [13] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A Survey on Aerial Swarm Robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, Aug 2018.
- [14] Y. Tian, K. Liu, K. Ok, L. Tran, D. Allen, N. Roy, and J. P. How, "Search and rescue under the forest canopy using multiple UAVs," *The International Journal of Robotics Research*, vol. 39, no. 10–11, pp. 1201–1221, 2020.

- [15] K. Miyano, R. Shinkuma, N. B. Mandayam, T. Sato, and E. Oki, "Utility based scheduling for multi-UAV search systems in disaster-hit areas," *IEEE Access*, vol. 7, pp. 26 810–26 820, 2019.
- [16] J. Sandino, F. Vanegas, F. Maire, P. Caccetta, C. Sanderson, and F. Gonzalez, "UAV framework for autonomous onboard navigation and people/object detection in cluttered indoor environments," *Remote Sensing*, vol. 12, no. 20, 2020.
- [17] W. Youn, H. Ko, H. Choi, I. Choi, J. H. Baek, and H. Myung, "Collision-free Autonomous Navigation of A Small UAV Using Low-cost Sensors in GPS-denied Environments," *International Journal of Control, Automation and Systems*, vol. 19, no. 2, pp. 953–968, 2021.
- [18] T. Dang, F. Mascarich, S. Khattak, H. Nguyen, H. Nguyen, S. Hirsh, R. Reinhart, C. Papachristos, and K. Alexis, "Autonomous Search for Underground Mine Rescue Using Aerial Robots," in *IEEE Aerospace Conference*. IEEE, mar 2020, pp. 1–8.
- [19] H. Goforth and S. Lucey, "GPS-Denied UAV Localization using Pre-existing Satellite Imagery," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 2974–2980.
- [20] F. Vanegas Alvarez and L. F. Gonzalez, "Enabling UAV navigation with sensor and environmental uncertainty in cluttered and GPS-denied environments," *Sensors*, vol. 16, no. 5, 2016.
- [21] J. Capitán Fernández, M. T. Spaan, L. Merino, and A. Ollero, "Decentralized Multi-Robot Cooperation with Auctioned POMDPs," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 650–671, may 2013.
- [22] F. Vanegas, K. J. Gaston, J. Roberts, and F. Gonzalez, "A Framework for UAV Navigation and Exploration in GPS-Denied Environments," in *IEEE Aerospace Conference*, Mar 2019, pp. 1–6.
- [23] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," *arXiv:1904.12901*, 2019.
- [24] X. Zhu, F. Vanegas, and F. Gonzalez, "An Approach for Multi-UAV System Navigation and Target Finding in Cluttered Environments," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, Sep 2020, pp. 1113–1120.
- [25] N. Bulling, "A Survey of Multi-Agent Decision Making," *KI - Künstliche Intelligenz*, vol. 28, no. 3, pp. 147–158, 2014.
- [26] W. Burgard, D. Fox, and S. Thrun, *Probabilistic Robotics*. The MIT Press, 2005.
- [27] F. A. Oliehoek and C. Amato, *A Concise Introduction to Decentralized POMDPs*, ser. SpringerBriefs in Intelligent Systems. Cham: Springer International Publishing, 2016.
- [28] S. S. Ponda, L. B. Johnson, A. Geramifard, and J. P. How, "Cooperative Mission Planning for Multi-UAV Teams," in *Handbook of Unmanned Aerial Vehicles*, K. P. Valavanis and G. J. Vachtsevanos, Eds. Dordrecht: Springer Netherlands, 2015, pp. 1447–1490.
- [29] C. V. Goldman and S. Zilberstein, "Decentralized control of cooperative systems: Categorization and complexity analysis," *Journal of Artificial Intelligence Research*, vol. 22, pp. 143–174, 2004.
- [30] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein, "The Complexity of Decentralized Control of Markov Decision Processes," *Mathematics of Operations Research*, vol. 27, no. 4, pp. 819–840, 2002.
- [31] D. S. Bernstein, "Complexity analysis and optimal algorithms for decentralized decision making," Ph.D. dissertation, University of Massachusetts Amherst, 2005.
- [32] D. Klimenko, J. Song, and H. Kurniawati, "TAPIR: A software Toolkit for approximating and adapting POMDP solutions online," in *Australasian Conference on Robotics and Automation (ACRA)*, 2014.
- [33] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, May 1998.
- [34] L. Meier, D. Honegger, and M. Pollefeys, "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6235–6240.
- [35] F. J. Perez-Grau, R. Ragel, F. Caballero, A. Viguria, and A. Ollero, "An architecture for robust UAV navigation in GPS-denied areas," *Journal of Field Robotics*, vol. 35, no. 1, pp. 121–145, 2018.