Mutual Relative Localization in Heterogeneous Air-ground Robot Teams

Samet Güler[a], İ. Emre Yıldırım[b] and H. Halid Alabay[c]
Dept. Electrical and Electronics Eng., Abdullah Gül University, Barbaros, Kayseri, 38080, Turkey

Keywords: Heterogeneous Multi-robot Systems, Relative Localization, Bayesian Filtering.

Abstract: Air and ground robots with distinct sensing characteristics can be combined in a team to accomplish demanding tasks robustly. A key challenge in such heterogeneous systems is the design of a local positioning methodology where each robot estimates its location with respect to its neighbors. We propose a filtering-based relative localization algorithm for air-ground teams composed of vertical-take-off-and-landing drones and unmanned aerial vehicles. The team members interact through a sensing/communication mechanism relying on onboard units, which results in a mutual connection between the air and ground components. Exploiting the supplementary features of omnidirectional distance sensors and monocular cameras, the framework can function in all environments without fixed infrastructures. Various simulation and experiment results verify the competency of our approach.

1 INTRODUCTION

Heterogeneous multi-robot systems (HMRS) offer unparalleled advantages in various tasks which require a collection of distinct sensing and manipulation capabilities such as area coverage, search and rescue, collaborative transportation, and signal source localization (Kushleyev et al., 2013; Staub et al., 2017; Tokekar et al., 2016; Manyam et al., 2016; Grocholsky et al., 2006; Vidal et al., 2002; Yu et al., 2015). In a HMRS, each individual robot acquires a local sensory information from the environment with its unique sensing capabilities. For instance, for a mapping objective, a set of aerial robots can collect visual representation of the environment while the local occupancy properties can be sensed by a group of ground robots. Then, the acquired data can be fused online to build a unified map, improving the efficiency and robustness compared to the homogeneous MRS approaches (Li et al., 2016; Mueggler et al., 2014; Kaslin et al., 2016).

Estimation of relative positions between team members in a MRS has been an active research area in the last decade. Traditional localization methods for HMRS rely on the existence of infrastructures which provide an absolute frame such as GPS and motion capture (mocap) systems. Such approaches lose validity in several real-life conditions, e.g., in GNSS-denied environments or rooms without mocap systems. To remove this fundamental necessity, the relative quantities between the team members should be estimated with the onboard perception capabilities. A common approach is to fuse distance sensors with proprioceptive onboard sensors such as IMU (Kia et al., 2016; Wallar et al., 2018; Prorok et al., 2011; Hepp et al., 2016; Kim and Kim, 2013; Nguyen et al., 2021). Since most distance sensors (e.g., ultrasonic and laser-based) provide one-directional measurement data with limited FOV, a collection of those sensors are implemented on the robots or in the environment to enlarge the operation zone of the robots in a MRS (Kim and Kim, 2013). On the other hand, ultrawideband (UWB) sensors offer unparalleled advantages in mobile robot localization thanks to their omnidirectional distance measurement with high accuracy and precision (Hepp et al., 2016; Wallar et al., 2018). However, one may lose the observability of

Figure 1: A sample heterogeneous team composed of five UGVs and a drone.

[a] https://orcid.org/0000-0002-9870-166X
[b] https://orcid.org/0000-0001-8035-0569
[c] https://orcid.org/0000-0001-5360-3655

Güler, S., Yıldırım, İ. and Alabay, H.
Mutual Relative Localization in Heterogeneous Air-ground Robot Teams.
DOI: 10.5220/0011267800003271
In *Proceedings of the 19th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2022)*, pages 304-312
ISBN: 978-989-758-585-2; ISSN: 2184-2809

the relative position with the distance only approaches in leader-follower formations (van der Helm et al., 2019).

Several works employed vision-based approach for relative localization in MRSs. Early works in this direction used tags with specific patterns mounted on robots to discriminate the robots from the background in camera images (Krajník et al., 2014; Saska et al., 2016; Roelofsen et al., 2015). With the recent developments in computer vision methods and computational units, the time required to detect sophisticated object models has reduced significantly, which enabled onboard detection of drones with deep learning techniques (Vrba and Saska, 2020). Although high detection rates for onboard drone detection using convolutional neural network (CNN) methods are reported in several tasks, the distance estimation mechanisms show poor performance with vision-only approaches. These results call for the need for an integration of the distance and vision sensors for reliable relative localization.

We address the relative localization objective in an air-ground team which consists of a team of drones and unmanned ground vehicles (UGV). We develop a filtering algorithm to estimate online the relative positions between the UGVs by utilizing the perception feedback of the drones. Our algorithm relies on the mutual interaction between the drone and the UGVs, established by the robots' onboard perception and local inter-robot communication. Particularly, the drones are commanded to hover around the designated locations on the UGV formation to provide the UGVs with the bearing measurements required to perform estimation. We performed several simulations and experiments with varying number of drones and UGVs. The main contributions of the work are as follows. We propose a complete localization and coordination framework for an air-ground robot team by integrating the onboard distance and vision sensing capabilities. Second, we propose a multi-rate extended Kalman filtering (EKF) algorithm which manages three measurement cases which occur as a result of the varying formation geometry of the HMRS. Finally, we illustrate several simulation and experiment results for the externally commanded UGVs case and formation control case. Our work differs from the previous HMRS applications (Staub et al., 2017; Tokekar et al., 2016; Grocholsky et al., 2006; Manyam et al., 2016; Vidal et al., 2002; Yu et al., 2015) in two aspects. First, the main objectives in the aforementioned works are path planning, map building or target tracking objectives with several infrastructure aids. Since our ultimate goal is to build safe and efficient swarms with arbitrary number of robots, we aim at improving the situational awareness of the robots in the swarm and focus on relative localization. Second, we utilize communication and sensor fusion by incorporating a visual sensor, UWB distance sensors, and IMUs onboard of the robots for an improved coordination between the air and ground entities. In (Cognetti et al., 2014), a drone equipped with vision sensors localizes a team of UGVs without any onboard tags exploiting the vertical motion capability of the drone. However, drones may lose tracking of the UGVs in an operation. In our framework, UWB sensors on board of the UGVs adds resiliency to the HMRS by pursuing the localization when the drones lose the UGVs in their image views.

The rest of the work is organized as follows. In Section 2, we define the HMRS model and formulate the localization problem. In Section 3, we propose the filtering and motion control algorithms. Section 4 demonstrates the simulation and experimental results as well as a discussion on the applications. Finally, Section 5 concludes the work.

2 OBJECTIVE

We consider a heterogeneous multi-robot system (HMRS) of $N+M+1$ robots, comprised of $N+1$ non-holonomic UGVs R_i, $i \in \{0,\dots,N\}$ and M drones D_j, $j \in \{1,\dots,M\}$. The robots are desired to move in coordination to accomplish a given objective. We focus on the design of a practical coordination mechanism between the robots. Designing such mechanisms in the absence of an infrastructure such as motion capture (mocap) systems remains a big challenge. Our goal is to devise a scalable and robust localization algorithm for the swarm that enables the realization of the typical practical distributed formation control algorithms.

To realize such a scheme without the aid of a GPS or mocap system, each robot should utilize its onboard capabilities and achieve a certain level of situational awareness. We assume that each UGV R_i is equipped with UWB sensors to acquire the inter-UGV distance measurements. Consider the common motion model for a non-holonomic UGV R_i as follows:

$$\mathrm{p}^i_{k+1} = \mathrm{p}^i_k + \left[\cos(\theta^i_k), \sin(\theta^i_k)\right]^\top v^i_k T_s, \tag{1}$$

$$\theta^i_{k+1} = \theta^i_k + \omega^i_k T_s, \tag{2}$$

where $\mathrm{p}^i = \left[x^i, y^i\right]^\top \in \Re^2$ is the 2D position, θ^i is the heading angle, v^i, ω^i are the linear and angular speed commands of R_i, $i \in \{0,\dots,N\}$, $k \in \mathbb{N}_+$ is the time step, and T_s is the sampling time constant.

We assume that each drone $\mathsf{D_i}$ is equipped with a monocular camera together with a flight control unit for the low-level control. Assuming that the motion in the roll and pitch axes are stabilized by the low-level controller, we denote by $\zeta = \left[x_d^i, y_d^i, \psi^i\right]^\top \in \Re^3$ the state vector of a drone $\mathsf{D_i}$, where x_d^i, y_d^i are the 2D coordinates and ψ^i is the heading angle with respect to a fixed frame $\mathcal{F}_\mathrm{G}$.

Our main objective is to design a robust localization algorithm for the system components such that each robot will be able to estimate or receive by communication the relative quantities required for a coordinated motion. We define the localization objective as follows. A UGV $\mathsf{R_i}$, $i = (1,\ldots,N)$ should estimate the relative positions $\mathbf{r}_i$ to its neighbors defined by an underlying constraint graph in the UGV team in the fixed frame $\mathcal{F}_\mathrm{G}$. Simultaneously, the drones should move in coordination with the UGVs and provide them with a perspective view and absolute sensor measurements to be used in the relative position estimations. Therefore, we aim at designing a Bayesian filtering algorithm and a formation control algorithm for the UGVs so that they can estimate their relative positions to their neighbors to move in coordination.

3 APPROACH

Our approach is divided in two parts below. First, we design the localization algorithm for the UGVs stating the assumptions on the drones' motion. Second, we present the drones' formation control mechanisms.

3.1 Ground Robot Localization

We assign UGV R_0 as the leader which moves with external commands or based on a path planning algorithm. Typically, the leader UGV is equipped with a better sensor suite than the other UGVs and determines suitable paths or waypoints for the entire team. For instance, it can build a collision-free path for the team with an onboard LIDAR sensor by assuming the entire team constructs a disc with certain radius. Noting that various path planning algorithms can be integrated, we focus on the design of our relative localization framework below.

Each follower UGV, $\mathsf{R_i}$, $i \in \{1,...,N\}$, uses the inter-robot distance measurements and the feedback signal received from the drones $\mathsf{D_i}$ to estimate its relative position $\mathbf{r}^i$ towards R_0. Assuming that all UGVs move on a 2D plane, we define the relative position system model between R_0 and $\mathsf{R_i}$, $i \in \{1,...,N\}$, as follows:

$$\mathbf{x}_{k+1}^i = \mathbf{x}_k^i + \begin{bmatrix} \mathbf{u}_k^3 \cos(\mathbf{x}_k^4) - \mathbf{u}_k^1 \cos(\mathbf{x}_k^3) \\ \mathbf{u}_k^3 \sin(\mathbf{x}_k^4) - \mathbf{u}_k^1 \sin(\mathbf{x}_k^3) \\ \mathbf{u}_k^2 \\ \mathbf{u}_k^4 \end{bmatrix} T_s, \qquad (3)$$

where $\mathbf{x}^i = \left[(\mathbf{r}^i)^\top, \theta^i, \theta^0\right]^\top$ is the state vector, $\mathbf{u} = \left[v^i, \omega^i, v^0, \omega^0\right]^\top$ is the input vector for the two-robot system $\{\mathsf{R}_0, \mathsf{R_i}\}$, and $\mathbf{r}^i = \left[r_x^i, r_y^i\right]^\top \in \Re^2$ is the relative position seen by $\mathsf{R_i}$ in the fixed frame $\mathcal{F}_\mathrm{G}$.

Furthermore, the objective of a drone $\mathsf{D_j}$, $j \in \{1,...,M\}$, is to detect and follow the pair $\{\mathsf{R}_0, \mathsf{R_i}\}$ where $i \in \{1,...,N\}$ (Fig. 2). Drone $\mathsf{D_j}$ changes the index i in its leader list $\{\mathsf{R}_0, \mathsf{R_i}\}$ with a certain time period such as $(\{\mathsf{R}_0, \mathsf{R}_1\}, \{\mathsf{R}_0, \mathsf{R}_2\}, \ldots, \{\mathsf{R}_0, \mathsf{R}_{N-1}\}, \{\mathsf{R}_0, \mathsf{R}_1\}, \ldots)$ based on a schedule which will be designed in Section 3.2. During an operation, all drones fly with a fixed heading in the fixed frame $\mathcal{F}_\mathrm{G}$, i.e., $\psi_k^i = \bar{\psi}$ for some arbitrary constant $\bar{\psi}$. Thus, the constant orientation of the drones serves as the absolute coordinate frame provider for the entire MRS. That is, the $x-y$ plane of the drone's body coordinate frame is utilized as the global frame. To maintain the constant heading during an operation, the drones may utilize their onboard magnetometers or a set of patterns in the environment. For instance, if the team is intended to operate in a certain environment with low magnetic deviation, the drones may trust their magnetometers which are available in most commercial flight control units.

Denote the bearing angles between R_0 and $\mathsf{R_i}$, $i \in \{1,...,N\}$, by $\phi^\mathrm{i} = \mathrm{atan}(r_y^i, r_x^i)$. A drone $\mathsf{D_i}$ aims at calculating and broadcasting the bearing angles ϕ^i and heading angles θ^i of the UGVs. For this purpose, the ground robots are equipped with certain patterns over-

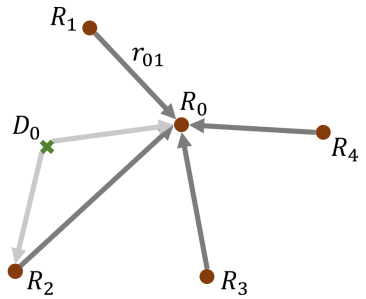

Figure 2: The proposed localization graph among a team of five UGVs and a drone projected onto a 2D plane. UGV R_0 serves as the leader for the team while each of the other UGVs $\mathsf{R_i}$, $i \in \{1,...,N\}$, acts as the first follower and detects and estimates their relative positions toward R_0. Each drone $\mathsf{D_j}$, $j \in \{1,...,M\}$, acts as the second follower and estimates its relative positions toward R_0 and a follower $\mathsf{R_i}$, $i \in \{1,...,N\}$ at any time.

head, e.g., qr code, so that their heading and bearing angles can be acquired by the drone's camera (Fig. 1). For convenience, drone $\mathsf{D_i}$ is assumed to fly at a sufficiently high altitude $\bar{z}$ so that it can detect a pair of robots $\{\mathsf{R_0}, \mathsf{R_i}\}$ by its camera. Such a suitable margin for $\bar{z}$ can be found empirically. Moreover, each follower UGV $\mathsf{R_i}$, $i \in \{1,...,N\}$, measures the distances $d_{0\mathrm{i}} = \|\mathbf{r}^i\|$ and the inter-UGV distances $d_{\mathrm{ij}} = \|\mathbf{r}^{ij}\| = \|\mathrm{p}^i - \mathrm{p}^j\|$ from the onboard UWB sensors. Therefore, the observed variables are the distances $d_{0\mathrm{i}}$, the heading angles θ^i, and the bearing angles ϕ^{i}.

To estimate $\mathbf{r}^i$ by robot $\mathsf{R_i}$, $i \in (1,...,N)$, we use a multi-rate EKF algorithm with three measurement models. Based on the availability of the measurements broadcast by drone $\mathsf{D_i}$, robot $\mathsf{R_i}$ performs the update stage using one of the following models:

$$\mathbf{y}^i = \begin{cases} \left[d_{0\mathrm{i}}, \phi^{\mathrm{i}}, \theta^i, \theta^0\right]^\top, & \text{if sw} = 1 \\ \left[d_{0\mathrm{i}}, \bar{\phi}^{\mathrm{i}}, \theta^0\right]^\top, & \text{if sw} = 2 \\ d_{0\mathrm{i}}, & \text{if sw} = 3 \end{cases} \tag{4}$$

where the switching signal sw denotes the drone's measurement mode. Denote the set of the pairs of UGVs detected by the drones at any time instant by $\mathcal{S} = \{\ldots\{\mathsf{R_0}, \mathsf{R_i}\}\ldots\}$, $i \in \{1,...,N\}$. Then, the measurement model of UGV $\mathsf{R_i}$, $i \in \{1,...,N\}$, at that time instant is determined based on the following rules:

- (sw=1) If $\{\mathsf{R_0}, \mathsf{R_i}\} \in \mathcal{S}$, we set $\text{sw} = i$, which implies the bearing ϕ^{i} is measured. Thus, the detected robot $\mathsf{R_i}$ uses the first measurement model which includes the distance $d_{0\mathrm{i}}$ and bearing ϕ^{i} toward $\mathsf{R_0}$, the self heading θ^i, and the heading θ^0 of $\mathsf{R_0}$.
- (sw=2) If $\{\mathsf{R_0}, \mathsf{R_i}\} \notin \mathcal{S}$ and $\mathcal{S} \neq \emptyset$, the UGV $\mathsf{R_i}$ uses the second measurement model where the bearing $\bar{\phi}^{\mathrm{i}}$ toward $\mathsf{R_0}$ is calculated artificially by using the formation geometry as follows. Consider Fig. 3 where two UGVs $\mathsf{R_j}, \mathsf{R_i}$ follow a leader UGV $\mathsf{R_0}$. Denote the internal angles formed by the triangles $\mathcal{T}(\mathsf{R_0}, \mathsf{R_j}, \mathsf{R_i})$ and $\mathcal{T}(\mathsf{R_0}, \mathsf{R_i}, \mathsf{R_j})$ by α_1 and α_2, respectively. By assumption, we exclude the collinear robots case, thus α_1, α_2 are well-defined. If $\{\mathsf{R_0}, \mathsf{R_j}\} \in \mathcal{S}$ and thus β_j is measured at a given time instant, then the bearing angle β_i for all robots $\mathsf{R_i}, i \neq j$ that are on the right half plane of the line l_i (the green-shaded area in Fig. 3) can be calculated as follows:

$$\beta_i = \beta_j + \pi - \alpha_1 - \alpha_2,$$

where

$$\alpha_1 = \arctan(\sqrt{1-\gamma^2}, \gamma), \quad \gamma = \frac{d_{0\mathrm{j}}^2 + d_{\mathrm{ij}}^2 - d_{0\mathrm{i}}^2}{2 d_{0\mathrm{j}} d_{\mathrm{ij}}}.$$

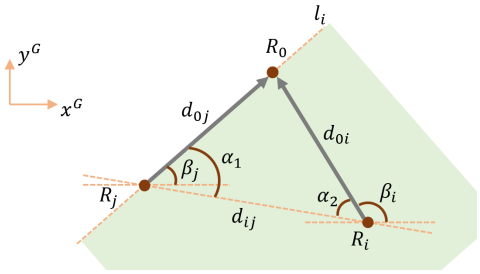

Figure 3: Calculation of the bearing angles for the UGVs not detected by the drones.

Similarly, α_2 can be calculated. Likewise, the bearing angles which lie on the left-half-plane of the line l_i can be calculated using the same method.

- (sw=3) If $\mathcal{S} \equiv \emptyset$, i.e., none of the drones can see a UGV pair $\{\mathsf{R_0}, \mathsf{R_i}\}$ for any $i \in \{1,...,N\}$, then all drones broadcast a Null message, meaning that all UGVs perform the iteration with the third measurement model by using the UWB distances only.

A pseudocode for the estimation algorithm of robot $\mathsf{R_i}$ is given in Algorithm 1. This algorithm harmonizes the sensing, communication, and filtering capabilities of the robots. To summarize, each UGV performs the usual EKF prediction step followed by the selection of the measurement model at the time instant k which determines the update step. Particularly, the linearized measurement model matrix is updated based on the measurement model of the UGV. We omit the details of the EKF algorithm for the sake of space, which can be found in (Thrun et al., 2005). In the proposed system, each UGV $\mathsf{R_i}$ estimates its relative position to $\mathsf{R_0}$ independently. In this sense, our application differs from the collaborative localization in which all team members update the same state vector estimate in a centralized manner.

Algorithm 1: EKF for robot $\mathsf{R_i}$.

Require: Previous state and covariance estimate
Ensure: Current state and covariance estimate
1: Prediction step
2: **if** Measurement model 1 **then**
3: $\quad y_k = \left[d_k^{0i}, \phi_k^i, \theta_k^i, \theta_k^0\right]$
4: **else if** Measurement model 2 **then**
5: $\quad y_k = \left[d_k^{0i}, \bar{\phi}_k^i, \theta_k^0\right]$
6: **else if** Measurement model 3 **then**
7: $\quad y_k = d_k^{0i}$
8: **end if**
9: Update step
10: $\hat{\mathbf{x}}_k \leftarrow$ New state estimate
11: $\Sigma_k \leftarrow$ New covariance estimate

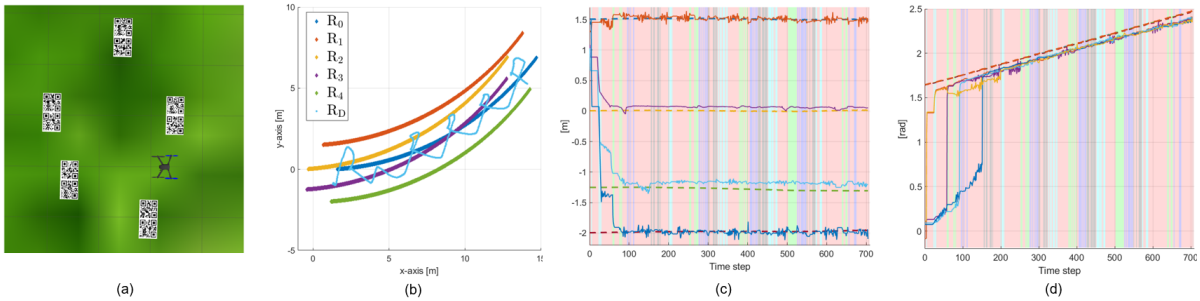

Figure 4: (a) A top view of the simulation setup; (b) traces of the robots; (c) relative position estimate; (d) heading angle estimations.

3.2 Drone Localization and Control

Each drone $\mathsf{D_j}$, $j \in \{1,...,M\}$ estimates its planar position with respect to the UGVs by using its onboard camera sensor. As the second follower in the underlying graph, a drone $\mathsf{D_j}$ aims at hovering on the midpoints m_i between the pair of UGVs $\{\mathsf{R}_0, \mathsf{R_i}\}$, $i \in \{1,...,N\}$ with respect to the following schedule. At the initial time step $k = 0$, the drones are commanded to track the midpoints $m_1, m_2, \ldots, m_M$. To provide every $\mathsf{R_i}$ with its bearing toward R_0, it is desired to visit all midpoints m_i, $i \in \{1,...,N\}$ periodically by the drones. We denote the time spent at a midpoint by k_m and set it as a design constant. Therefore, at the time instants $k = bk_m$, $b \in \mathbb{N}_+$, the drones switch their commanded midpoints to $m_{b+1}, m_{b+2}, \ldots, m_{b+M}$ until $N = b+M$, where N is the total number of the UGVs. When $N = b+M+1$, the commanded midpoints for the drones is switched back to $m_1, m_2, \ldots, m_M$.

When hovering on the midpoints m_i, the drones uses computer vision techniques to detect the special patterns on the UGVs. When the drone detects a pair $\{\mathsf{R}_0, \mathsf{R_i}\}$, it calculates and broadcasts the heading angles θ_0, θ_i and the bearing angle ϕ_i and sets the parameter sw accordingly. Remarkably, since the drone maintains a constant heading $\bar{\psi}$ during an operation, the bearing angle ϕ_i is calculated with respect to $\bar{\psi}$ which serves as the absolute north for the entire team. Furthermore, the drones use the desired formation geometry to calculate the control signal required to move between the midpoints. In the next section, we apply the typical proportional controller to control the drones' motions on the horizontal plane, noting that more advanced control algorithms can be applied to control the drones' motions with higher precision.

4 RESULTS

We evaluated the proposed framework both in simulations and real experiments. We aimed at demonstrating the applicability of the framework and its integration into formation control algorithms.

4.1 Simulations

We used a Gazebo environment including Rosbot UGVs and Iris drones integrated to the ROS interface. We mounted two qr code patterns on each Rosbot to be able to detect its heading angle by a mono camera attached at the bottom of the drone. The UGVs were accepting linear and angular velocity inputs and placed in a hexagon formation with the leader (R_0) in the front (Fig. 4a). Each follower UGV performed relative position estimation independently obeying the state and observation models given in Section 3.1. The drone implemented the px4 software which accept the high-level planar velocity and yaw commands by handling the low-level control. All robots ran their estimation code at 20 Hz. We acquired the inter-UGV distances from the simulation model, emulating the UWB distance measurements.

We show the traces of the robots and the relative position and heading estimation results of a simulation in Fig. 4. In this simulation, the UGVs moved on an arc with constant linear and angular velocities, and the drone followed the midpoints periodically for $t_m = 8$ seconds per robot with the sequence $(m_4, m_1, m_2, m_3, m_4, \ldots)$ by utilizing a custom control algorithm. In Fig. 4b and Fig. 4c, the green background shows the time instants where the drone observes the pair $\{\mathsf{R}_0, \mathsf{R}_1\}$, the blue background for $\{\mathsf{R}_0, \mathsf{R}_2\}$, the black background for $\{\mathsf{R}_0, \mathsf{R}_3\}$, and the cyan background for $\{\mathsf{R}_0, \mathsf{R}_4\}$. The UGV R_4 corrected its relative position estimation to the leader (green colored robot) successfully when the drone detected the robot pair $\{\mathsf{R}_0, \mathsf{R}_4\}$ and broadcast ϕ_4 at the start of the operation (Fig. 4c). Meanwhile, the other UGVs $\mathsf{R}_1, \mathsf{R}_2$, and R_3 calculated their bearing angles ϕ_1, ϕ_2, and ϕ_3, respectively, by using the formation geometry (as given in Section 3.1) and corrected their relative position estimations accordingly.

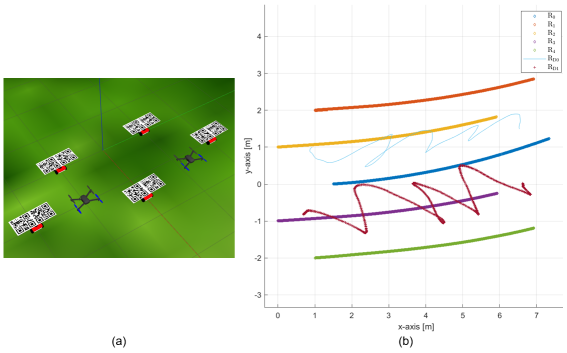

Figure 5: (a) A simulation setup with two drones; (b) Paths of the robots in an arc motion test.

On the other hand, every follower UGV corrects its own heading estimation once the drone visits its corresponding midpoint (Fig. 4d). The main reason for the undetected time zones (red background) was the switching motion of the drone between the midpoints. Remarkably, although the total time without UGV detection was almost half of the simulation time, the framework showed high performance in estimating the actual relative positions.

Next, we used two drones and five UGVs in 11 simulations to demonstrate the scalability of our approach (Fig. 5). The first drone is assigned to detect the UGVs $\mathsf{R}_0, \mathsf{R}_1$, and R_2, and the second drone aims at detecting the UGVs $\mathsf{R}_0, \mathsf{R}_3$, and R_4. The follower UGVs R_i, $i \in \{1,...,4\}$, used the EKF outcome to maintain their desired relative positions toward the leader R_0 in a hexagon formation. We define the estimation root mean square errors (RMSE) by

$$e_i^j = \left[\frac{1}{K} \sum_{k=1}^{K} \left(\hat{\mathbf{r}}_k^{ij} - \mathbf{r}_k^{ij} \right)^2 \right]^{\frac{1}{2}}, \qquad (5)$$

where K is the total time steps during an experiment, $\hat{\mathbf{r}}_1^i, \hat{\mathbf{r}}_2^i$ are the relative pose estimations of the ith UGV's EKF for the x and y axes, respectively, with $j \equiv \{x, y\}$ denoting the coordinate axis. We observed that e_i^j remained below 0.1 meters for all i, j in all tests. We did not observe significant performance variation with the change in the drone's follow time t_m. Compared to the first set of simulations, the lower estimation errors stemmed from the fact that each drone is responsible for two UGVs, which increased the time duration spent on top of each UGV.

4.2 Experiments

The experimental setup consisted of a custom-built drone with an F450 frame and three non-holonomic UGVs (two Rosbots and a Turtlebot) (Fig. 6). A Rosbot was assigned as the leader (R_0), and the other

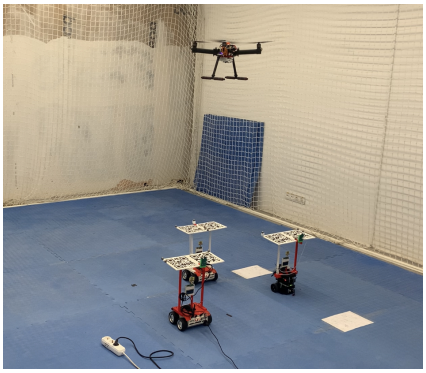

Figure 6: The experimental setup consisted of a custom-built drone of diameter 45cm equipped with a Pixhawk flight controller together with two Rosbot and a Turtlebot UGVs.

UGVs were to estimate the relative positions $\mathbf{r}_1, \mathbf{r}_2$. Each UGV was equipped with calibrated Decawave ultrawideband distance sensors at the center and two qr code patterns on the top. A Raspberry Pi4 with a camera performed UGV detection and high-level computations on the drone while the low-level control was handled by a Pixhawk 4 flight controller. We collected the ground truth position data and controlled the drone's low-level motion with a Vicon motion capture system. We used the traditional computer vision techniques and packages to detect the qr code patterns. We observed that the camera detected reliably the patterns at the distance interval $1.4 < h < 1.85$ meters, and thus we flew the drone at the altitude $h = 1.75$ m.

In the first set of experiments, the UGVs were steered with exogenous velocity commands, and the drone was controlled to follow the midpoints m_1, m_2. An experiment's result is given in Fig. 7 where the UGVs followed a straight path with $v_i = 0.05$ m/s and $\omega_i = 0$ rad/s. The drone was commanded to follow each midpoint for $t_m = 4$ seconds before switching to the other midpoint. In Fig. 7b and Fig. 7c, the green background shows the time instants where the drone observes both R_0 and R_1, the blue background for both R_0 and R_2, and the black background for all $\mathsf{R}_0, \mathsf{R}_1$ and R_2. Finally, the light red background shows that the drone cannot detect any UGV. Once the drone detected the pair $\{\mathsf{R}_0, \mathsf{R}_1\}$ for the first time (the first green time instant), the EKF of UGV R_1 uses the first measurement model in (4) which corresponds to the locally observable instant, and the correct relative position and heading $\hat{\theta}_1$ are estimated. However, $\hat{\theta}_2$ could not be estimated until the drone could detect the heading of R_2 at the time step 400 (the first time instant the blue background is seen in Fig. 7d). Notably, this time instant corresponds to the unobservable mode, which we aim at studying in our future

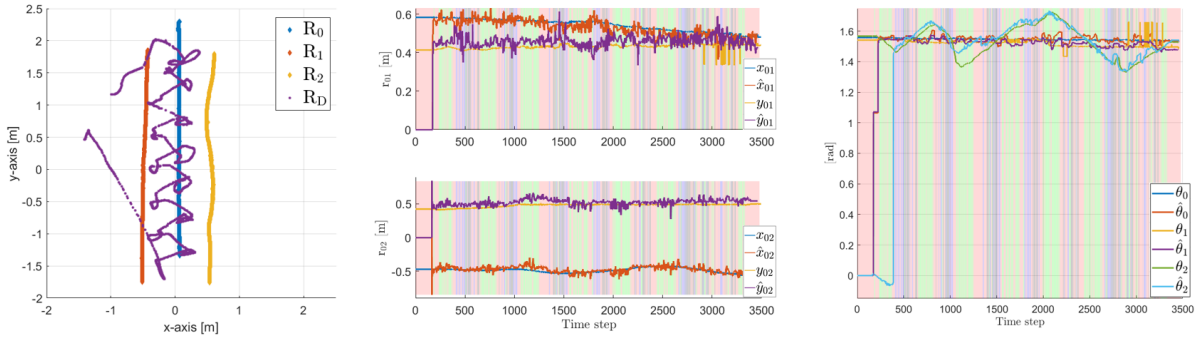

Figure 7: Results of an experiment with externally commanded UGVs: (a) Traces of the robots; (b) relative position estimates $\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2$; (c) heading angle estimations $\hat{\theta}_i$. Background color codes: Green: $\{\mathsf{R}_0, \mathsf{R}_1\}$ detected; Blue: $\{\mathsf{R}_0, \mathsf{R}_2\}$ detected; Black: $\{\mathsf{R}_0, \mathsf{R}_1, \mathsf{R}_2\}$ detected; Red: No pair detected.

works. We emphasize that the relative position and heading estimations were initiated far from their correct values in this experiment. Nevertheless, the filter succeeded to approach the correct values. Table 1 shows the RMSE for the relative position estimations (5) in five experiments. We observed that the average error remained below 0.16 meters in all experiments. We note that the drone's high-level controller was tuned to follow smoothly the midpoints m_1, m_2.

In the second set of experiments, we used the perception layer output as feedback to a formation control layer. Particularly, the follower UGVs tried to maintain the desired relative positions $\mathbf{r}_1^{\text{des}}, \mathbf{r}_2^{\text{des}}$ toward the leader R_0 by using the estimation outcomes $\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2$. We illustrate a formation control experiment in Fig. 8 and Fig. 9 where R_1 and R_2 use custom control algorithms to maintain the desired relative poses $\mathbf{r}_1^{\text{des}} = [0.5, 0.5]^\top, \mathbf{r}_2^{\text{des}} = [-0.5, 0.5]^\top$ m toward the leader R_0 which is commanded externally to follow an S-shaped path. Similar to the previous case, the drone was controlled to switch between the midpoints m_1 and m_2. We observed that the estimation performance sufficed to maintain the desired formation as long as the drone tracks the midpoints smoothly. We provide further insights and a detailed discussion in the next part.

In Fig 10, we show the RMSE for the formation errors in eight experiments, which are defined as follows:

$$f_{i,j} = \left[\frac{1}{K} \sum_{k=1}^{K} \left(\mathbf{r}_k^{ij} - \mathbf{r}^{ij,\text{des}} \right)^2 \right]^{\frac{1}{2}},$$

where K is the total time steps during an experiment, i is the follower UGV identity, and $j \equiv \{x, y\}$ denotes

Table 1: Relative position estimation RMSEs of five experiments in meters.

	Experiment No				
	1	**2**	**3**	**4**	**5**
e_1^x	0.133	0.132	0.135	0.104	0.157
e_1^y	0.099	0.188	0.115	0.089	0.137
e_2^x	0.109	0.098	0.105	0.106	0.135
e_2^y	0.110	0.191	0.115	0.120	0.155

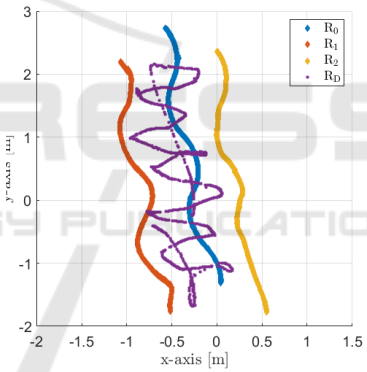

Figure 8: Traces of the robots in a formation control experiment.

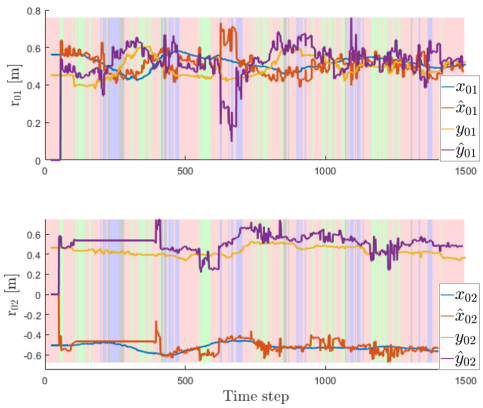

Figure 9: Relative position estimates $\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2$ in a formation control experiment. Background color codes: Green: $\{\mathsf{R}_0, \mathsf{R}_1\}$ detected; Blue: $\{\mathsf{R}_0, \mathsf{R}_2\}$ detected; Black: $\{\mathsf{R}_0, \mathsf{R}_1, \mathsf{R}_2\}$ detected; Red: No pair detected.

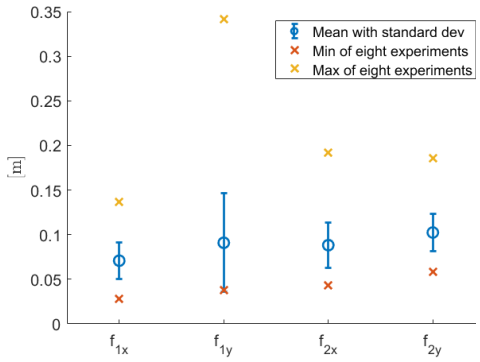

Figure 10: RMSE of eight formation control experiments.

the coordinate axis. Although we did not observe a direct correlation between the error levels and the coordinate axes or robots' identities, we observed that the average error in each axis remained below 0.12 meters, which proves the efficiency of the proposed method.

We emphasize the unique difficulty of the relative localization problem in a HMRS, particularly when combined with a formation control algorithm. In contrast to the traditional localization objective where a set of anchors located at known locations serve as beacons to find the location of a group of tag sensors, relative localization systems try to generate hypothesis in the absence of an anchor. For this reason, the relative position estimators usually cannot generate centimeter level accuracy as in the case of the traditional estimators (Vrba and Saska, 2020). Nevertheless, we observed satisfactory results both in the localization and formation control simulations and experiments.

To overcome the issue of deviating drones due to unobserved patterns, we developed a custom control algorithm for the UGVs that make use of the leader velocity, which proved useful when the formation moved on arcs. In our primary tests, the drone tended to deviate from the midpoints of the UGV pairs, which we overcame by designing a control law for the drone that utilized the single UGV detection as well. Therefore, we emphasize that a designer should fine tune the drone controller before implementing the proposed framework.

5 CONCLUSIONS AND FUTURE WORK

We have addressed the relative localization in heterogeneous multi-robot systems, a key challenge for the realization of such systems in a variety of applications. We have proposed a Bayesian filter which estimates the relative positions among the UGVs in certain configurations by utilizing their onboard sensing and communication mechanisms together with the drones' perception. We have demonstrated through simulations and experiments that the proposed method yields sufficient estimation performance under the case of intermittent perception feedback acquired from the aerial vehicle. A comprehensive set of simulations and experiments have proven that the estimation outcome yield sufficient accuracy for the formation maintenance without external infrastructures.

Future work may include a realization of the proposed model with more UGVs and drones both indoors and outdoors. Also, a more robust control algorithm design for the drone can improve the estimation performance. Furthermore, we aim at carrying out a detailed observation analysis for the system model to investigate the observable and unobservable modes for the varying cases of drone detection.

ACKNOWLEDGEMENTS

This paper has been produced benefiting from the 2232 International Fellowship for Outstanding Researchers Program of TÜBİTAK (Project No: 118C348). However, the entire responsibility of the paper belongs to the owner of the paper. The financial support received from TÜBİTAK does not mean that the content of the publication is approved in a scientific sense by TÜBİTAK.

REFERENCES

Cognetti, M., Oriolo, G., Peliti, P., Rosa, L., and Stegagno, P. (2014). Cooperative control of a heterogeneous multi-robot system based on relative localization. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.

Grocholsky, B., Keller, J., Kumar, V., and Pappas, G. (2006). Cooperative air and ground surveillance. *IEEE Robotics & Automation Magazine*, 13(3):16–25.

Hepp, B., Nägeli, T., and Hilliges, O. (2016). Omnidirectional person tracking on a flying robot using occlusion-robust ultra-wideband signals. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 189–194.

Kaslin, R., Fankhauser, P., Stumm, E., Taylor, Z., Mueggler, E., Delmerico, J., Scaramuzza, D., Siegwart, R., and Hutter, M. (2016). Collaborative localization of aerial and ground robots through elevation maps. In *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE.

Kia, S. S., Rounds, S., and Martinez, S. (2016). Cooperative localization for mobile agents: A recursive decentralized algorithm based on kalman-filter decoupling. *IEEE Control Systems*, 36(2):86–101.

Kim, S. J. and Kim, B. K. (2013). Dynamic ultrasonic hybrid localization system for indoor mobile robots. *IEEE Transactions on Industrial Electronics*, 60(10):4562–4573.

Krajník, T., Nitsche, M., Faigl, J., Vaněk, P., Saska, M., Přeučil, L., Duckett, T., and Mejail, M. (2014). A practical multirobot localization system. *Journal of Intelligent & Robotic Systems*, 76(3-4):539–562.

Kushleyev, A., Mellinger, D., Powers, C., and Kumar, V. (2013). Towards a swarm of agile micro quadrotors. *Autonomous Robots*, 35(4):287–300.

Li, J., Deng, G., Luo, C., Lin, Q., Yan, Q., and Ming, Z. (2016). A hybrid path planning method in unmanned air/ground vehicle (UAV/UGV) cooperative systems. *IEEE Transactions on Vehicular Technology*, 65(12):9585–9596.

Manyam, S. G., Casbeer, D. W., and Sundar, K. (2016). Path planning for cooperative routing of air-ground vehicles. In *2016 American Control Conference (ACC)*. IEEE.

Mueggler, E., Faessler, M., Fontana, F., and Scaramuzza, D. (2014). Aerial-guided navigation of a ground robot among movable obstacles. In *2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)*, pages 1–8.

Nguyen, T. H., Nguyen, T.-M., and Xie, L. (2021). Range-focused fusion of camera-IMU-UWB for accurate and drift-reduced localization. *IEEE Robotics and Automation Letters*, 6(2):1678–1685.

Prorok, A., Tomé, P., and Martinoli, A. (2011). Accommodation of nlos for ultra-wideband tdoa localization in single- and multi-robot systems. In *2011 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–9.

Roelofsen, S., Gillet, D., and Martinoli, A. (2015). Reciprocal collision avoidance for quadrotors using on-board visual detection. IEEE.

Saska, M., Baca, T., Thomas, J., Chudoba, J., Preucil, L., Krajnik, T., Faigl, J., Loianno, G., and Kumar, V. (2016). System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization. *Autonomous Robots*, 41(4):919–944.

Staub, N., Mohammadi, M., Bicego, D., Prattichizzo, D., and Franchi, A. (2017). Towards robotic MAGMaS: Multiple aerial-ground manipulator systems. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*. MIT press.

Tokekar, P., Hook, J. V., Mulla, D., and Isler, V. (2016). Sensor planning for a symbiotic UAV and UGV system for precision agriculture. *IEEE Transactions on Robotics*, 32(6):1498–1511.

van der Helm, S., Coppola, M., McGuire, K. N., and de Croon, G. C. H. E. (2019). On-board range-based relative localization for micro air vehicles in indoor leader–follower flight. *Autonomous Robots*, 44(3-4):415–441.

Vidal, R., Shakernia, O., Kim, H., Shim, D., and Sastry, S. (2002). Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *IEEE Transactions on Robotics and Automation*, 18(5):662–669.

Vrba, M. and Saska, M. (2020). Marker-less micro aerial vehicle detection and localization using convolutional neural networks. *IEEE Robotics and Automation Letters*, 5(2):2459–2466.

Wallar, A., Araki, B., Chang, R., Alonso-Mora, J., and Rus, D. (2018). Foresight: Remote sensing for autonomous vehicles using a small unmanned aerial vehicle. In Hutter, M. and Siegwart, R., editors, *Field and Service Robotics*, pages 591–604, Cham. Springer International Publishing.

Yu, H., Meier, K., Argyle, M., and Beard, R. W. (2015). Cooperative path planning for target tracking in urban environments using unmanned air and ground vehicles. *IEEE/ASME Transactions on Mechatronics*, 20(2):541–552.