Preprints of the 23rd IFAC World Congress
Busan, Republic of Korea, August 23-28, 2026

# Experimental Validation of Resilient Multi-UAV Control against Agent Failures

**Takuya Murakami** * **Toru Namerikawa** **

* *School of Integrated Design Engineering, Keio University, Yokohama, Japan (e-mail: murakamit@keio.jp).*
** *Department of System Design Engineering, Keio University, Yokohama, Japan (e-mail: namerikawa@sd.keio.ac.jp).*

**Abstract:** This paper proposes a resilient formation control framework for leader-follower multi-UAV systems subject to non-compensable actuator faults. We develop a scheme that extends resilient consensus theory to second-order linear dynamics in three-dimensional space, integrated with a coordinated leader replacement mechanism. Through numerical simulations and indoor flight experiments, we demonstrate that non-faulty agents maintain formation via autonomous leader replacement, even in the presence of severe faults. These results confirm that multi-UAV formations can recover from non-compensable failures.

*Keywords:* Multi-agent systems, Resilient control, Formation control, UAVs, Quadrotor, Experimental validation

## 1. INTRODUCTION

The deployment of Unmanned Aerial Vehicles (UAVs) for public surveillance and smart logistics is highly anticipated for realizing smart cities (Dúbravová et al., 2024). To expand their utility, Multi-Agent Systems (MAS) of UAVs have attracted growing interest for cooperative missions (Skorobogatov et al., 2020). While experimental demonstrations are advancing (Zhang et al., 2025), most fault-tolerant control methods assume compensable faults (Davoodi et al., 2016; Wang and Dong, 2024), where the faulty agent can still be stabilized. Non-compensable faults, such as hard-over or float failures that override the actuator output, can propagate through the network and destabilize the entire system.

Leader-follower formation control is widely adopted for its simplicity (Li et al., 2013; Kotani et al., 2021). However, the leader constitutes a single point of failure. Although resilience against adversarial followers has been studied (Rezaee et al., 2021; Li et al., 2024; Wang et al., 2024), multi-leader approaches by Usevitch and Panagou (2022) require leader-level redundancy, and leader replacement schemes by Li et al. (2017) typically assume all followers remain healthy. Few results address non-compensable faults affecting both leaders and followers under second-order dynamics with real-platform validation.

Building on our prior work Murakami and Namerikawa (2025a,b), this paper proposes a resilient control framework to maintain precise formation and tracking stability. This framework (i) tolerates non-compensable faults in a leader-follower network with second-order dynamics, and (ii) autonomously replaces the leader upon failure. The main contribution is the extension from planar settings to three-dimensional flight. This extension addresses the system integration challenges of realizing theoretical decentralized control laws on physical multi-UAV platforms in a 3D environment. To achieve this, we developed an indoor multi-UAV testbed incorporating independent altitude regulation to validate the resilient mechanism in hardware. The theoretical stability analysis for this control law is provided in Murakami and Namerikawa (2026).

## 2. PROBLEM FORMULATION

### 2.1 UAV Model and Control Objective

Modern low-level flight controllers decouple and linearize the translational motion of quadrotors, enabling high-level control design on simplified models (Kotani et al., 2021). We model $N$ UAVs in three-dimensional space as second-order linear systems with acceleration inputs:

$$\begin{cases} \dot{p}_i(t) = v_i(t), \\ \dot{v}_i(t) = u_i(t), \quad (i = 1, 2, \ldots, N), \end{cases} \tag{1}$$

where $p_i(t) = [x_i \ y_i \ z_i]^\top \in \mathbb{R}^3$, $v_i(t) \in \mathbb{R}^3$, and $u_i(t) \in \mathbb{R}^3$ denote the position, velocity, and control input of agent $i$.

The control objective is to design a distributed law such that all non-faulty agents achieve $p_i(t) \to p_L(t) + d_i$ as $t \to \infty$, even under non-compensable actuator faults.

### 2.2 Graph Notation and Assumptions

The communication topology is represented by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with adjacency matrix $\mathcal{A} = [a_{ij}]$,

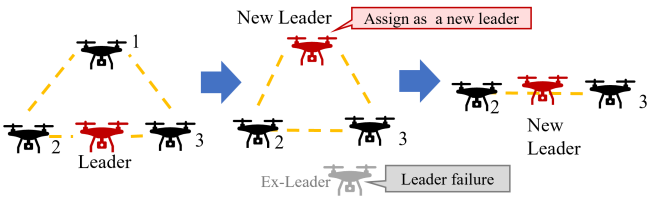



Fig. 1. Concept of the proposed approach.

©2026 the authors. Accepted by IFAC for publication under a Creative Commons Licence CC-BY-NC-ND

where $a_{ij} = 1$ if agent $i$ receives information from agent $j$. The Laplacian is $\mathcal{L} = \mathcal{D} - \mathcal{A}$, with $\mathcal{D}$ the degree matrix. An additional node $N+1$ represents the leader's target position provided by a human operator.

We assume: (i) after faults, non-faulty followers remain connected to the leader, and (ii) the communication network is undirected. Assumption (i) is a standard requirement in consensus-based resilient control; it ensures that the non-faulty subgraph retains sufficient connectivity for the control law to converge. Relaxing this assumption, e.g. through minimum dwell-time analysis of the switching topology, is left for future work.

## 3. FAULT MODEL

A fault is termed *non-compensable* when the actuator output is overridden by a signal independent of the controller, so that no control redesign can correct it. Following Boskovic and Mehra (2003); Ducard (2009), we categorize actuator faults into four classes. Let $u_i^f(t)$ denote the actual input under failure:

$$u_i^f(t) = \begin{cases} u_{\text{sat}}, & \text{(a) Hard-over,} \\ \sigma\, u_i(t), & \text{(b) Loss of effectiveness,} \\ u_i(t_f), & \text{(c) Lock-in-place,} \\ \mathbf{0}, & \text{(d) Float,} \end{cases} \tag{2}$$

where $u_{\text{sat}}$ is the actuator saturation limit, $\sigma \in (0,1)$ the effectiveness coefficient, and $t_f$ the fault occurrence time. Among these, types (a), (c), and (d) are non-compensable. The temporal profiles are shown in Fig. 2.

Hard-over faults are particularly critical, as they can destabilize the entire formation (Ducard, 2009). In the experiments, faults are injected into the horizontal inputs ($x$-$y$ plane) while altitude control remains nominal. We assume noise-free state measurements in both simulation and experiments.

## 4. PROPOSED CONTROL LAW

### 4.1 Formation Control

We propose a distributed control law for agent $i$ formulated as

$$u_i = -\sum_{j=1}^{N+1} w_{ij}\Big[k_1(h_i - h_j) + k_2(\dot{h}_i - \dot{h}_j)\Big]. \tag{3}$$

Here $h_i \in \mathbb{R}^3$ is the formation error variable defined by $h_i = p_i - l_i d_i$, where $d_i \in \mathbb{R}^3$ is the desired offset from the leader and $l_i$ is the leader-follower indicator ($l_i = 0$ for the leader, $l_i = 1$ for followers). The scalars $k_1, k_2 > 0$ are feedback gains. To account for agent reliability, the weighted adjacency element $w_{ij}(t)$ is defined as

$$w_{ij}(t) = a_{ij}\,\eta_i(t)\,\eta_j(t), \tag{4}$$

where $\eta_i(t) \in [0,1]$ represents the trust value of agent $i$.

The leader aims to reach the target position $p_{N+1}$ while broadcasting the desired relative target position $d_i$ to the followers.

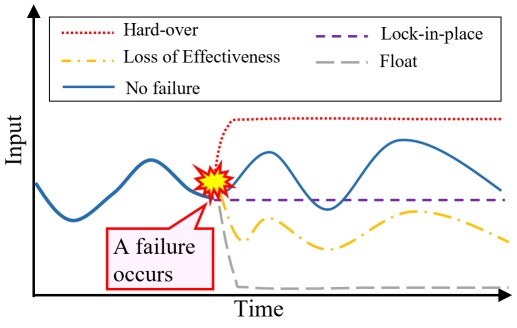



Fig. 2. Temporal profiles of four actuator failure modes.

### 4.2 Fault Detection and Reliability

A distributed observer is used to calculate residuals:

$$\ddot{\hat{p}}_i = -\sum_{j=1}^{N+1} w_{ij}\Big[k_1(\hat{h}_i - \hat{h}_j) + k_2(\dot{\hat{h}}_i - \dot{\hat{h}}_j)\Big] + m\,\epsilon_i, \tag{5}$$

where $m \in \mathbb{R}$ is the observer gain, $\hat{p}_i \in \mathbb{R}^3$ denotes the estimated position of agent $i$, and $\hat{h}_i = \hat{p}_i - l_i d_i$ is the estimated formation error. The residual vector $\epsilon_i \in \mathbb{R}^3$ is computed as

$$\epsilon_i(t) = (p_i - \hat{p}_i) + \sum_{j=1}^{N+1} w_{ij}\,(p_j - \hat{p}_j). \tag{6}$$

Fault detection is performed by comparing the residual norm $\|\epsilon_i\|$ with a threshold $\tau$. Based on the detection result, the trust value $\eta_i(t)$ and the associated disappointment variable $\delta_i(t)$ are defined by

$$\dot{\delta}_i = -\alpha\,\delta_i(t) + (1 - e^{-\beta t_i})\,\rho_i(t), \tag{7}$$

$$\eta_i(t) = \max(1 - \delta_i,\, 0), \tag{8}$$

where $t_i$ denotes the duration of the current threshold $\|\epsilon_i\|$ violation, and $\rho_i(t)$ is the switching function defined as $\rho_i(t) = 1$ if $\|\epsilon_i\| > \tau$ and $\rho_i(t) = 0$ otherwise.

### 4.3 Leader Replacement Procedure

To choose a new leader, we prioritize the agent whose motion is most consistent with the target trajectory. Specifically, the agent whose velocity vector is most aligned with the target velocity $\dot{p}_{N+1}$ is chosen to ensure the smooth transition during the handover.

Let $\mathcal{N}_L$ denote the set of followers adjacent to the current leader. For each neighbor $j \in \mathcal{N}_L$, we calculate an alignment score based on the cosine similarity, defined as

$$\cos(\theta_j) = \frac{\dot{p}_{N+1} \cdot \dot{p}_j}{\|\dot{p}_{N+1}\|\|\dot{p}_j\| + 1}. \tag{9}$$

Based on these calculated values, the new leader $j_{\max}$ is determined by selecting the agent with the maximum alignment score from the neighboring set:

$$j_{\max} = \arg\max_{j \in \mathcal{N}_L} \cos(\theta_j). \tag{10}$$

The proposed control law enables non-faulty agents to maintain their desired relative positions $d_i$ with respect to the leader. Since each agent uses only data from its neighbors, the computational cost per agent scales with the number of neighbors and remains low even for large networks.

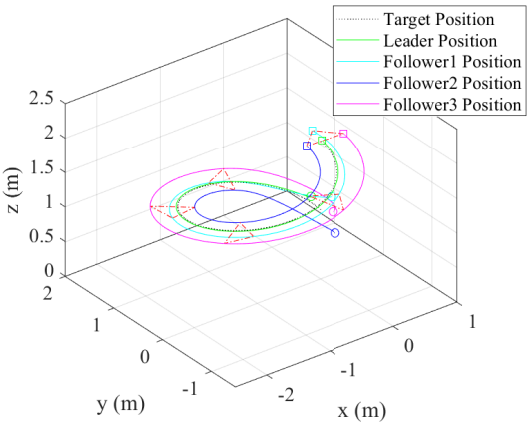



Fig. 3. Nominal 3D helical trajectories without failure (dashed: target).

## 5. SIMULATION RESULTS

### *5.1 Simulation Setup*

We evaluate the system with $N = 4$ agents. The target trajectory is a 3D helix with a radius of 1 m in the $x$-$y$ plane and a height of 1 m along the $z$-axis over 10 s. The leader tracks this target, while the three followers maintain a rigid formation with respect to the leader. The simulation parameters are listed in Table 1. Three cases are considered: (1) nominal (no fault), (2) leader hard-over fault without the proposed control, and (3) leader hard-over fault with the proposed control.

Table 1. Parameters for simulations

| Parameter | Symbol | Value |
|---|---|---|
| Position gain | $k_1$ | 15 |
| Velocity gain | $k_2$ | 5 |
| Distributed observer gain | $m$ | 3 |
| Fault detection threshold | $\tau$ | 0.2 |
| Recovery rate | $\alpha$ | 0.1 |
| Fault sensitivity | $\beta$ | 0.2 |
| Hard-over failure signal | $u_{\text{sat}}$ | $[-0.05, 0.05]^\top$ |

### *5.2 Nominal Condition*

The trajectories under nominal conditions are shown in Fig. 3, where the dashed line indicates the target helix and the solid lines represent the agent paths (the leader in green, followers in skyblue, blue, and purple). The proposed controller enables the followers to track the leader along the helical trajectory while preserving the formation geometry. Both consensus errors and residuals remained negligible throughout the maneuver, which is consistent with the expected baseline behavior.

### *5.3 Performance under Leader Failure*

A hard-over fault is injected into the leader at $t = 2$ s to evaluate system resilience. Fig. 4 compares the 3D trajectories under the standard consensus control (left) and the proposed resilient framework (right). In both panels, the black dashed line denotes the leader's target

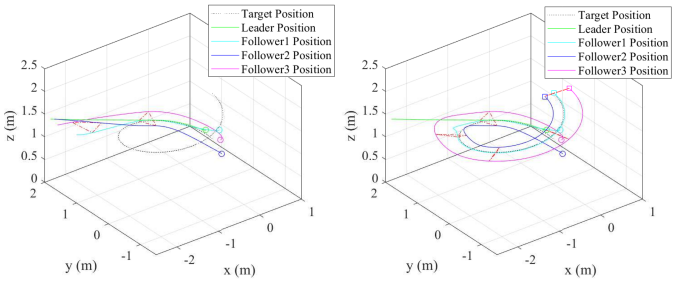



Fig. 4. Trajectory comparison under a leader hard-over fault. Left: standard consensus control. Right: proposed resilient control.

position and the solid lines represent the agent paths, where the faulty leader and follower 1 are plotted in green and skyblue, respectively.

Without the proposed isolation mechanism, the non-faulty followers continue to track the faulty leader due to the static communication topology, leading to formation divergence. In contrast, the proposed method allows the non-faulty followers to detect the fault and isolate the leader using the reliability weighting. Subsequently, follower 1 is elected as the new leader, enabling the remaining agents to maintain the formation. These results show that the proposed method can reconfigure the system against non-compensable leader faults.

## 6. EXPERIMENTAL VALIDATION

### *6.1 Experimental System*

As illustrated in Fig. 6, the experiments are conducted in an indoor laboratory equipped with a motion capture system.

*UAV Platform and Dynamics* The experimental platform consists of two quadrotors (DJI TELLO EDU). While the framework is scalable to $N$ agents, this fundamental leader-follower setup is sufficient to validate the resilience mechanisms within the confined indoor testbed.

Each quadrotor is equipped with an onboard attitude controller provided by the manufacturer. Since the specific mapping from the control commands (0–100) to the physical motion response is undocumented, we conducted characterization experiments in the motion-capture environment.

The velocity response to a constant command exhibited a constant-acceleration profile. Repeating this for various input values, we found that the resulting acceleration increases approximately linearly with the command value, as shown in Fig. 5. The slope of the least-squares fit was 0.009 m/s$^2$ per unit command, which we adopt as the scaling factor. Hence, the TELLO EDU can be modeled as a second-order system with acceleration inputs, consistent with (1).

In the formation control experiments, the altitude is regulated to approximately 1 m by the proportional controller, and the formation motion is restricted to the horizontal $x$–$y$ plane. Fault injection along the vertical axis was excluded because hard-over or float failures in altitude could

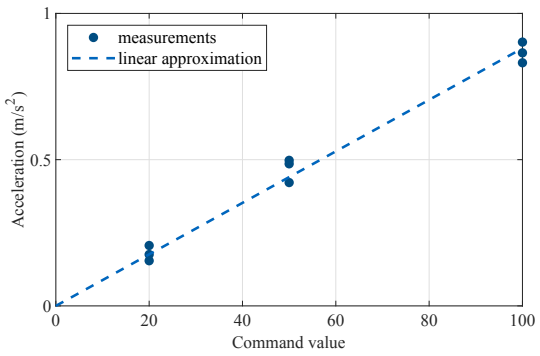



Fig. 5. Identified acceleration vs. command value.

cause the UAV to crash into the ground or the ceiling of the safety net, posing unacceptable risks; hence, the horizontal plane provides the most critical and safely evaluable setting for validating the fault-tolerance and leader replacement mechanisms. Based on the characterization above, the control inputs $u_i(t)$ calculated by the proposed controller are converted into device commands using the identified scaling factor. All flights are conducted indoors in a confined workspace enclosed by a collapsible safety net with approximate dimensions 3.0 m × 4.0 m × 2.0 m. Since positions and velocities are directly provided by the motion capture system, state observability is satisfied in this experimental setup.

*Sensing, Communication, and Computing* The positions and attitudes of the quadrotors are measured by an OptiTrack Flex 13 motion capture system (four cameras, 120 Hz sampling rate). The system streams a position vector $p_i(t) = (x_i, y_i, z_i)$ and an orientation quaternion $q_i(t)$ for each rigid body via UDP over a wired LAN to the control PC. On the control PC, the quaternion is converted to a yaw angle $\psi_i(t)$ for heading control. The control PC (Intel Core i5, 16 GB RAM) reads the pose at 10 Hz and transmits commands to the quadrotors over Wi-Fi (UDP). The tracking volume within the safety net is approximately 3 m × 2 m.

*Software Architecture* The control software is developed in Python 3 and organized into modules for motion capture reception, coordinate transformation, distributed formation control, and command transmission. The position control module rotates horizontal errors into the body frame using the yaw angle before generating commands. A proportional yaw controller keeps $\psi_i \approx 0°$. The leader's target position $p_{N+1}(t)$ is set by a human operator through keyboard inputs. Although executed on a single PC, the software emulates the distributed structure: each agent is a separate block that computes its control using only local and neighbor data.

### 6.2 Formation Control Validation

The experimental parameters are summarized in Table 2. After takeoff, the target position was shifted in four steps of approximately 0.9 m: backward ($y$ positive), left ($x$ negative), forward ($y$ negative), and right ($x$ positive). Throughout the experimental results, solid lines denote the measured UAV positions and dashed lines denote the target positions. As shown in Figs. 7 and 8, the quadrotors

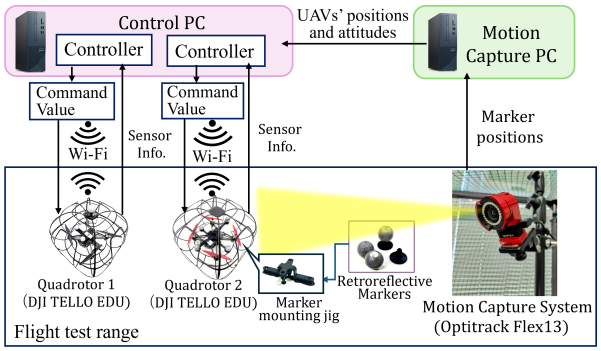



Fig. 6. Quadrotor flight experimental system overview.

Table 2. Parameters for Experiments

| Parameter | Symbol | Value |
|---|---|---|
| Control loop rate | — | 10 Hz |
| Mocap sampling rate | — | 120 Hz |
| Command-to-accel. scaling | — | 0.009 m/s$^2$ |
| Altitude setpoint | — | 1.0 m |
| Initial target position (m) | $p_L(0)$ | $[0\ 0\ 1]^\top$ |
| Initial relative position (m) | $d_2(0)$ | $[1\ 0\ 1]^\top$ |
| Position gain | $k_1$ | 1.4 |
| Velocity gain | $k_2$ | 0.6 |
| Yaw hold gain | $k_\psi$ | 20 |
| Observer gain | $m$ | 3 |
| Fault detection threshold | $\tau$ | 0.2 |
| Trust recovery rate | $\alpha$ | 0.1 |
| Fault sensitivity | $\beta$ | 0.2 |

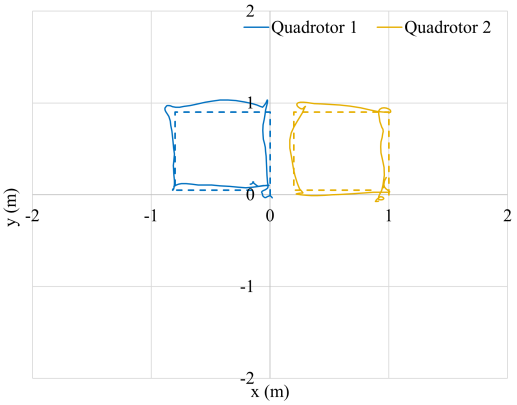



Fig. 7. Formation flight trajectories in the $x$-$y$ plane.

tracked the target while maintaining the formation geometry. Fig. 9 confirms that the residuals remained below the detection threshold, with no false positive detections.

### 6.3 Scenario 1: Hard-Over Failure

We use this scenario to examine leader replacement when a non-compensable fault occurs in the leader. The leader has a hard-over failure in which its control input in the $x$-axis is replaced by the maximum allowable value, as shown in Fig. 10. After the formation is established, a forward command in the positive $y$-direction is sent. Immediately after this command, the leader encounters the fault and starts drifting leftward in the negative $x$-direction.

The resulting trajectories are shown in Figs. 11a and 11b. The leader's data end at $t = 30$ s when it leaves

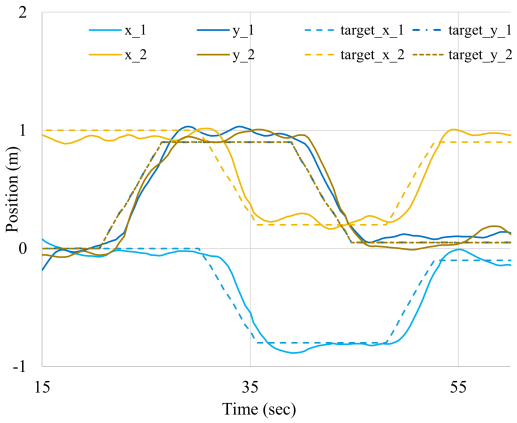



Fig. 8. Time histories of $x - y$ positions during flight.

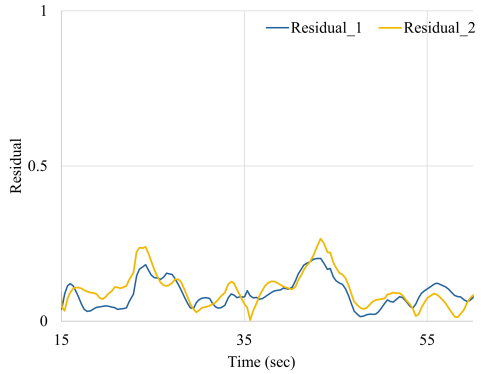



Fig. 9. Residuals during formation flight.

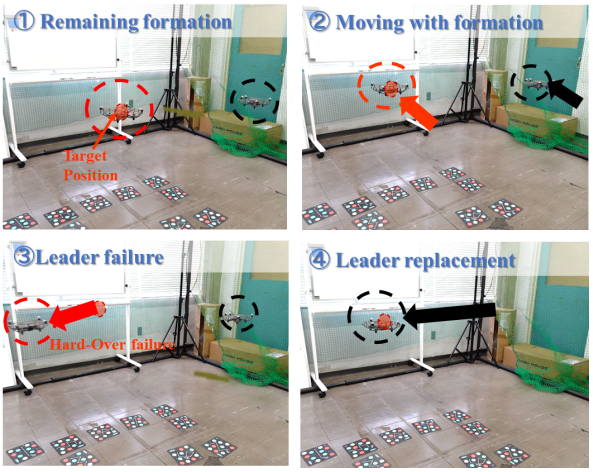
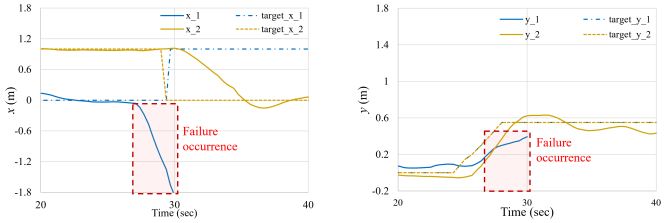



Fig. 10. Experiment scenario 1: hard-over failure.

the measurement area and collides with the safety net. Around $t = 26$ s, after the fault, the leader begins to drift in the $x$-direction, and the system changes the leader to quadrotor 2, showing that leader replacement has occurred. The corresponding residual signals are presented in Fig. 12. The residual increases after the fault, showing the effect of the leader fault. In this experiment, the proposed control law detected the leader fault, replaced the leader, and kept the formation.




(a) $x$-axis (b) $y$-axis

Fig. 11. Position time histories under hard-over failure.

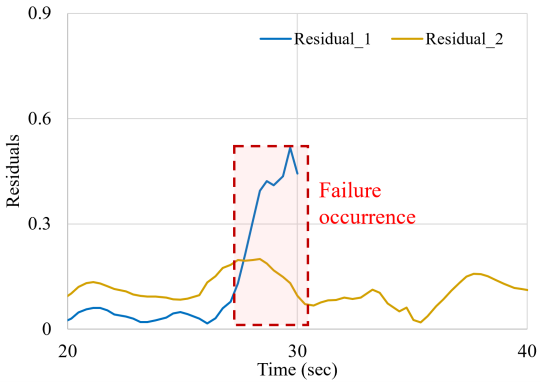



Fig. 12. Residuals under hard-over failure.

### *6.4 Scenario 2: Float Failure and Recovery*

We use this scenario to test both fault and recovery. It examines fault detection and leader switching during the fault period, and also whether the system can bring the recovered quadrotor back into the formation. As shown in Fig. 13, the leader (quadrotor 1) has a float fault between $t = 27$ s and $t = 41$ s while the two quadrotors fly in formation, and then recovers at $t = 40$ s.

The position time histories are shown in Figs. 14a and 14b. From the $x$-axis data, leader switching takes place at about $t = 33$ s: quadrotor 2 becomes the new leader, moves to the target, and converges. From the $y$-axis data, quadrotor 1 stays almost at the same position between $t = 27$ s and $t = 41$ s despite changes in the target due to the fault; after recovery at $t = 41$ s, it moves toward the target.

Fig. 15 shows the residuals during the fault experiment. The residual of the non-faulty quadrotor 2 stays below 0.2, while the residual of quadrotor 1 rises above the threshold $\tau = 0.2$ during the fault period, showing that the observer detects the fault as intended. In this scenario, the proposed control law detected the leader fault, switched the leader to maintain the formation, and later brought the recovered quadrotor back into the formation.

## 7. CONCLUSION

This paper presented a resilient control framework for leader-follower multi-UAV systems that combines trust-based fault isolation with autonomous leader replacement to maintain formation under non-compensable actuator faults.

Simulation results with four agents and hardware experiments with two quadrotors confirmed that the system

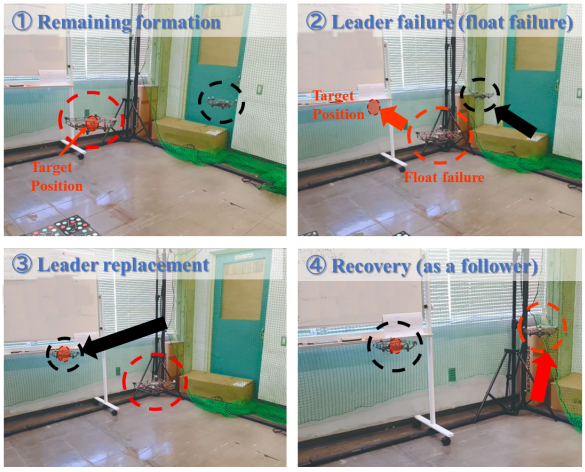
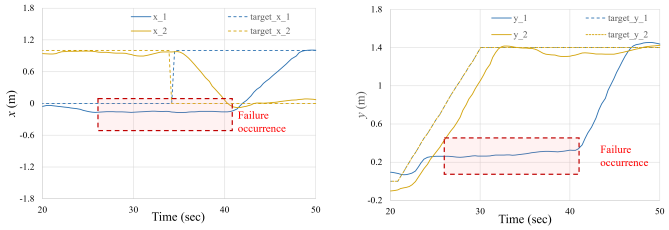



Fig. 13. Experiment scenario 2: float failure.




(a) $x$-axis (b) $y$-axis

Fig. 14. Position time histories under float failure.

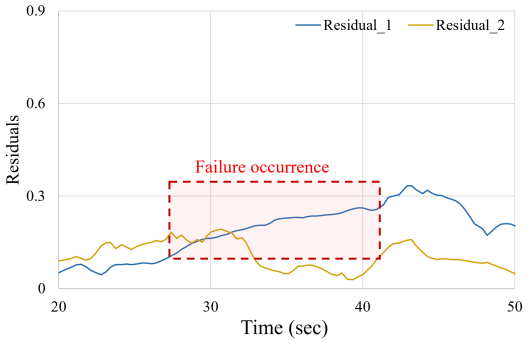



Fig. 15. Residuals under float failure.

detects severe leader faults, executes leader replacement, and preserves the formation. The experiments also demonstrated that a recovered agent can be reintegrated into the formation. Although the formation geometry inherently maintains inter-agent separation under nominal conditions, explicit collision avoidance during fault transients is not addressed in this paper. Integration with control barrier functions (CBFs) is planned for future open-field deployment.

Future work will extend this framework to sensor faults, cyber-physical attacks, and switching topologies to relax connectivity assumptions, further enhancing the resilience of multi-agent systems in contested environments.

## REFERENCES

Boskovic, J. and Mehra, R.K. (2003). Failure detection, identification and reconfiguration in flight control. In *Fault Diagnosis and Fault Tolerance for Mechatronic Systems: Recent Advances*, chapter 1, 129–167. Springer.

Davoodi, M.R., Meskin, N., and Khorasani, K. (2016). Simultaneous fault detection and control design for a network of multi-agent systems. *Automatica*, 66, 185–194.

Dúbravová, H., Bureš, V., and Velfl, L. (2024). Review of the application of drones for smart cities. *IET Smart Cities*, 6(4), 312–332.

Ducard, G.J.J. (2009). *Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*. Springer.

Kotani, K., Guo, Z., Namerikawa, T., and Qu, Z. (2021). Cooperative transport control by a multicopter system. *IET Control Theory and Applications*, 15(6), 861–876.

Li, F., Ding, T., Zhou, M., Hao, K., and Chen, L. (2017). An affection-based dynamic leader selection model for formation control in multirobot systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7), 1217–1228.

Li, J., Hua, Y., Yu, J., Dong, X., Feng, Z., and Ren, Z. (2024). Resilient time-varying formation tracking control for general linear multiagent systems with a nonautonomous leader and adversarial followers. *IEEE Transactions on Cybernetics*, 54(10), 6203–6216.

Li, Z., Liu, X., Ren, W., and Xie, L. (2013). Distributed tracking control for linear multiagent systems with a leader of bounded unknown input. *IEEE Transactions on Automatic Control*, 58(2), 518–523.

Murakami, T., and Namerikawa, T. (2025). Consensus control for resilient multi-agent systems considering leader failure. In *SICE International Symposium on Control Systems 2025*, 38–42. Osaka, Japan.

Murakami, T., and Namerikawa, T. (2025). Resilient formation control in multi-agent systems considering leader failure. *SICE Journal of Control, Measurement, and System Integration*, 18(1), 2510766.

Murakami, T., and Namerikawa, T. (2026). Resilient multi-UAV formation control with leader replacement under agent failures. *Mechatronics*, Vol. 115, 103465, 2026.

Rezaee, H., Parisini, T., and Polycarpou, M.M. (2021). Resiliency in dynamic leader-follower multiagent systems. *Automatica*, 125, 109384.

Skorobogatov, G., Barrado, C., and Salamí, E. (2020). Multiple UAV systems: A survey. *Unmanned Systems*, 8(2), 149–169.

Usevitch, J. and Panagou, D. (2022). Resilient trajectory propagation in multirobot networks. *IEEE Transactions on Robotics*, 38(1), 42–61.

Wang, Y., Han, L., Li, X., Hao, P., and Ren, Z. (2024). Resilient consensus for discrete-time multiagent systems with a dynamic leader and time delay: Theory and experiment. *IEEE Transactions on Cybernetics*, 54(2), 1015–1027.

Wang, H. and Dong, J. (2024). Robust fault-tolerant formation maneuver control for multiagent systems with mismatched disturbances. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(6), 3762–3773.

Zhang, S., So, O., Garg, K., and Fan, C. (2025). GCBF+: A neural graph control barrier function framework for distributed safe multiagent control. *IEEE Transactions on Robotics*, 41(2), 1533–1548.