2016 International Conference on
Unmanned Aircraft Systems (ICUAS)
June 7-10, 2016. Arlington, VA USA

Real-Time Fault-Tolerant Cooperative Control of Multiple UAVs-UGVs in the Presence of Actuator Faults

Mohamed A. Kamel[1], Khaled A. Ghamry[1], and Youmin Zhang[1]

***Abstract*— This paper investigates fault-tolerant cooperative control (FTCC) strategy for a team of unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) in the presence of actuator faults. When actuator faults occur in one or more of the UGVs, two cases are considered: 1) the faulty UGV cannot complete its assigned task due to a severe fault occurrence, it has to get out from the formation mission. Then, FTCC strategy is designed to re-assign the mission to the remaining healthy vehicles; and 2) the faulty UGV can continue the mission with degraded performance, then the other team members will reconfigure their controllers considering the capability of faulty UGV. Thus, the FTCC strategy is designed to re-coordinate the motion of each UAV-UGV in the team. FTCC problem is formulated as an optimal assignment problem, where a Hungarian algorithm is applied. Real-time experiments are presented in order to demonstrate the effectiveness of the proposed FTCC scheme in different fault scenarios.**

***Index Terms*— Fault-tolerant cooperative control, unmanned aerial vehicles, unmanned ground vehicles**

I. INTRODUCTION

In recent years, research in unmanned systems have attracted a great interest in both civilian and military applications. During mission execution, unmanned systems should maintain a pre-defined formation shape, avoid collision of obstacles and also other team members, and accommodate occurred faults and mitigate their negative effect on mission execution. Such an objective can be achieved by so-called fault-tolerant cooperative control (FTCC) strategies [1].

FTCC can be achieved with task re-assignment as well as motion re-coordination. The basic idea of FTCC is that if one or more UAVs/UGVs subject to a fault, then there are two cases: 1) One or more UAVs/UGVs subject to a severe fault, and are not able to accomplish their assigned task. So, the mission should be re-assigned to the remaining healthy vehicles; 2) One or more UAVs/UGVs subject to a fault. However they are still able to complete the mission but with the degraded performance. In this case, the other healthy UAVs-UGVs will reconfigure their controllers considering the capability of the faulty ones.

Although cooperative control in fault-free case is extensively studied in the literature, but FTCC has not been fully investigated in the literature (see [2] and the references therein). The existing studies mainly focus on communication faults [3], and the obstacle avoidance problem to avoid the collision with the faulty UAV [4]. In [5], the FTCC problem for a team of UAVs are considered. The formation recovery algorithm is proposed based on a trajectory re-planning technique. Once an actuator fault occurs, a formation supervisor commands all the UAVs to re-plan their trajectories within the physical constraints of the faulty vehicle. In authors' previous work [6], an FTCC scheme for a team of UGVs is designed in the fault-free case for performing the desired formation. Once a fault occurs, the faulty UGV gets out from the formation, and the healthy UGVs reconfigure the formation shape to complete the mission in a decentralized manner. Formation reconfiguration is based on the Graph Theory. In [7], an FTCC scheme is considered for multiple UAVs. Feasible references in response to actuator faults can be generated by considering the health status of the team. While the FTCC gains can converge within finite time to facilitate the fault accommodation by applying the auxiliary integrated regressor matrix and vector method. This work can be seen as a motion re-coordination technique to keep the formation in the presence of actuator fault in two types of unmanned systems (UAVs and UGVs). From the existing literature, determining how to accommodate the fault effect on the whole team during formation in the presence of severe actuator fault is still an open issue. Furthermore, most of the aforementioned references consider only the numerical simulations and have not yet validated in the real-time experiments.

Motivated by the aforementioned issue, a new FTCC algorithm for a team of UAVs-UGVs is presented in this paper. This work is an extension of authors' previous work [1] by considering both task re-assignment and motion re-coordination, and the real-time implementation of both faulty cases. The team consists of one UAV and N UGVs. The UAV assigned as the team leader, while the UGVs are the followers. Once an actuator fault occurs in UGVs, two cases can be considered: i) if the faulty UGV can no longer complete the mission due to severe fault occurrence, its communication module will be switched off and this UGV will be out from the formation. Subsequently, the other healthy UAV-UGVs can reconfigure their formation shape and continue the mission. Formation is reconfigured by solving the FTCC problem as an optimal assignment problem, ensuring that during formation reconfiguration one and only one UGV should be assigned to a unique place in the new formation shape; and ii) if the faulty UGV can still complete the mission with degraded performance, then the controllers of the other healthy UAV-UGVs are reconfigured by considering the reduced capability of the

[1]Mohamed A. Kamel, Khaled A. Ghamry, and Youmin Zhang are with the Department of Mechanical and Industrial Engineering, Concordia University, Montréal, QC H3G 1M8, Canada. m_atef@encs.concordia.ca, k_ghamry@encs.concordia.ca, youmin.zhang@concordia.ca (Corresponding author)

faulty UGV. This is implemented by regenerating a new reference trajectory that the leader UAV and the team must follow considering the reduced capabilities of the faulty UGV. As a result, the mission will be continued but with degraded performance. The main contributions of this work are: 1) developing an FTCC algorithm under both regular and severe actuator faults occurrence cases; and 2) real-time implementation and validation with a team of UAVs-UGVs testbeds available at the Networked Autonomous Vehicles Laboratory of Concordia University for the proposed control algorithm in different faulty situations.

The paper is hereinafter organized in the following sections. In Section II the description of the UAVs and UGVs, their control architecture and kinematic model are presented, the control objective is defined, and the formation problem is formulated. FTCC algorithm is developed in Section III. Section IV presents the experimental results for different fault cases, and conclusions of this work are presented in Section V.

II. PRELIMINARIES

The problem considered in this paper is to drive a team of UAVs-UGVs in a desired formation motion in faulty cases. To design an FTCC scheme, formation control scheme under fault-free cases must be designed first as a preparation to the fault occurrence cases. This section presents kinematics and modeling of the UAVs and UGVs, together with formation control of them under fault-free cases.

A. Formation Geometry

Consider j^{th} vehicle in a team of UAVs-UGVs moving in a specific formation within a leader-follower scheme, $j \in \{l, 1, 2, \ldots, N\}$ denotes the formation configuration of the leader l and N followers. The leader l should track a predefined trajectory $(x_r(t), y_r(t), z_r(t))$ defined in a time interval $t \in [0, T]$, while N followers should follow the leader for maintaining a desired formation configuration $l_d - \phi_d$ relative to the leader, where l_d and ϕ_d are the desired formation distance and angle for each follower with respect to the leader respectively. So, UAV-UGVs' controllers are constrained by the following condition:

$$\lim_{t\to\infty}((q_N(t) - q_l(t)) - F^d) = 0 \quad \forall\, N < j \tag{1}$$

where q_N and q_l are the current postures of the followers and the leader respectively. Eq. (1) means that each follower should maintain desired formation distances with respect to the leader.

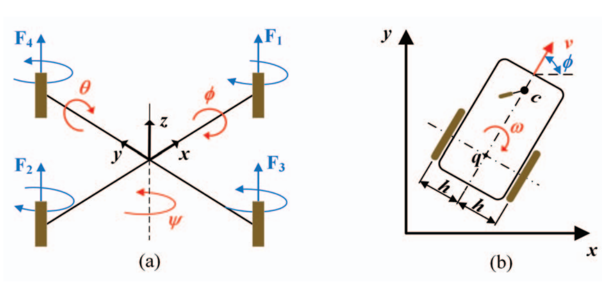

Fig. 1. (a) Coordination system of the quadrotor UAV. (b) Schematic diagram of the differentially-driven wheeled mobile robot

B. UAV Model

The UAV considered here is a quadrotor helicopter, where four rotors laid up symmetrically around its center as illustrated in Fig. 1a. The quadrotor dynamic model is obtained using the Newton-Euler formalism. The simplified quadrotor dynamics are:

$$\begin{aligned}
\ddot{x} &= (\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi)\frac{U_z}{M} & \ddot{\phi} &= \frac{U_\phi}{J_{xx}} \\
\ddot{y} &= (\sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi)\frac{U_z}{M} & \ddot{\theta} &= \frac{U_\theta}{J_{yy}} \\
\ddot{z} &= -g + (\cos\theta\cos\phi)\frac{U_z}{M} & \ddot{\psi} &= \frac{U_\psi}{J_{zz}}
\end{aligned} \tag{2}$$

where ϕ, θ, and ψ are the Euler angles which represent roll, pitch and yaw respectively. M is the quadrotor mass. J_{xx}, J_{yy}, and J_{zz} are the quadrotor moment of inertia according to x, y, and z axes respectively.

C. UGV Model

The UGVs considered in this work are unicycle mobile robots. The nonlinear kinematic equations of the robot is:

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 \\ \sin\phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{3}$$

where v and ω are the linear and angular velocities of the robot. The right and left velocities of the robot wheels are:

$$v_R = v + h\omega, \qquad v_L = v - h\omega \tag{4}$$

where h is the distance between the vehicle longitudinal axis and each wheel.

D. UAV Control

A combination of sliding mode control (SMC) and linear quadratic regulator (LQR) is used for UAV control as shown in Fig. 2. More details can be found in authors' previous work [8]. A brief explanation is presented here for convenience.

Due to nonholonomic features of a quadrotor, the dynamic model is divided into two subgroups: a fully actuated system represented by the following equation

$$\begin{bmatrix} \ddot{z} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{U_1}{M}\cos\theta\cos\phi - g \\ \frac{U_4}{J_{zz}} \end{bmatrix} \tag{5}$$

and an under-actuated subsystem defined as:

$$\begin{aligned}
\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} &= \frac{U_1}{M}\begin{bmatrix} \cos\psi & \sin\psi \\ \sin\psi & -\cos\psi \end{bmatrix}\begin{bmatrix} \sin\theta\cos\phi \\ \sin\phi \end{bmatrix} \\
\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \end{bmatrix} &= \begin{bmatrix} \frac{U_2}{J_{xx}} \\ \frac{U_3}{J_{yy}} \end{bmatrix}
\end{aligned} \tag{6}$$

The objective of the fully actuated subsystem controller is to minimize the error in the altitude e_z and yaw angle e_ψ respectively. To achieve this objective, SMC is applied to generate the control inputs U_1 and U_4.

Two loops are designed to control the under-actuated subsystem with an outer loop and an inner loop. The objective of the outer loop controller is to obtain the desired position in

x and y axes. This is achieved by design an LQR controller. While the objective of the inner loop controller is to converge the actual values of the Euler angles ϕ and θ to their desired values ϕ_r and θ_r obtained from the outer loop controller. SMC is applied to generate the control inputs U_2 and U_3 to satisfy accurate quadrotor attitude stabilization.

E. UGV Control

UGV control algorithm is based on a combination of feedback linearization and linear model predictive control (MPC) for each UGV in the team. The linearized model of each UGV with nonlinear dynamics is found through feedback linearization, while MPC is applied to the linear model to perform the motion control as well as to keep the formation shape. Details of this work can be found in [9]. A brief explanation will be presented here for convenience.

Nonholonomic robots are not input-state linearizable. However they are input-output linearizable. By carrying out the input-output feedback linearization steps, following linearized model can be obtained:

$$\ddot{z}_1 = u_1, \qquad \ddot{z}_2 = u_2 \tag{7}$$

where u_1 and u_2 are the new control inputs. The resulting dynamic compensator is then represented as:

$$\begin{aligned} \dot{\xi} &= u_1 \cos\phi + u_2 \sin\phi \\ v &= \xi \\ \omega &= \frac{u_2 \cos\phi - u_1 \sin\phi}{\xi} \end{aligned} \tag{8}$$

As shown in Fig. 2, the outputs of MPC are the optimum values of the control inputs u_1 and u_2. These signals should be fed to the dynamic compensator (8) in order to obtain the actual control inputs v and ω and consequently get the speeds of the right and left motors v_R and v_L as presented in (4).

III. FTCC ALGORITHM WITH COLLISION AVOIDANCE CAPABILITIES

The FTCC algorithm consists of 1) the FDD scheme in order to detect and isolate the actuator fault; 2) the task re-assignment and decision making algorithm in which the mission is re-assigned or re-coordinated based on the fault situation; and 3) the collision avoidance algorithm. The following assumptions are made for the FTCC algorithm.

Assumption 1: Each UAV-UGV in the formation has each own fault detection and diagnostics (FDD) algorithm. So it can detect the fault and estimate the value of the loss of effectiveness factor γ.

Assumption 2: Each UAV-UGV in the team receives the position of other team members, i.e. each vehicle knows the position of other vehicles in the formation.

Assumption 3: The UAV leader will not subject to a severe fault.

Assumption 4: There is no communications loss between the team members.

Assumption 5: In this study, the actuator fault is considered as a severe fault if the loss of effectiveness factor γ is 0.65 or higher.

A. Task Re-Assignment and Decision Making Algorithm

The basic idea of the FTCC algorithm is to deal with the actuator faults occurred in one or more UGVs according to the fault signal γ sent from the faulty UGV as shown in Fig. 3. The following equation represents the value of γ:

$$0 \le \gamma_j \le 1 \qquad j = 1, \ldots, N. \tag{9}$$

Depending on the value of γ_j, the following situations may take place:

- If If $\gamma_j = 0$, then all UGVs are fault-free. So, the whole team continue the planned mission.
- If $0.65 < \gamma_j < 1$, then one or more UGVs are subject to actuator fault. The faulty UGVs detected the fault,

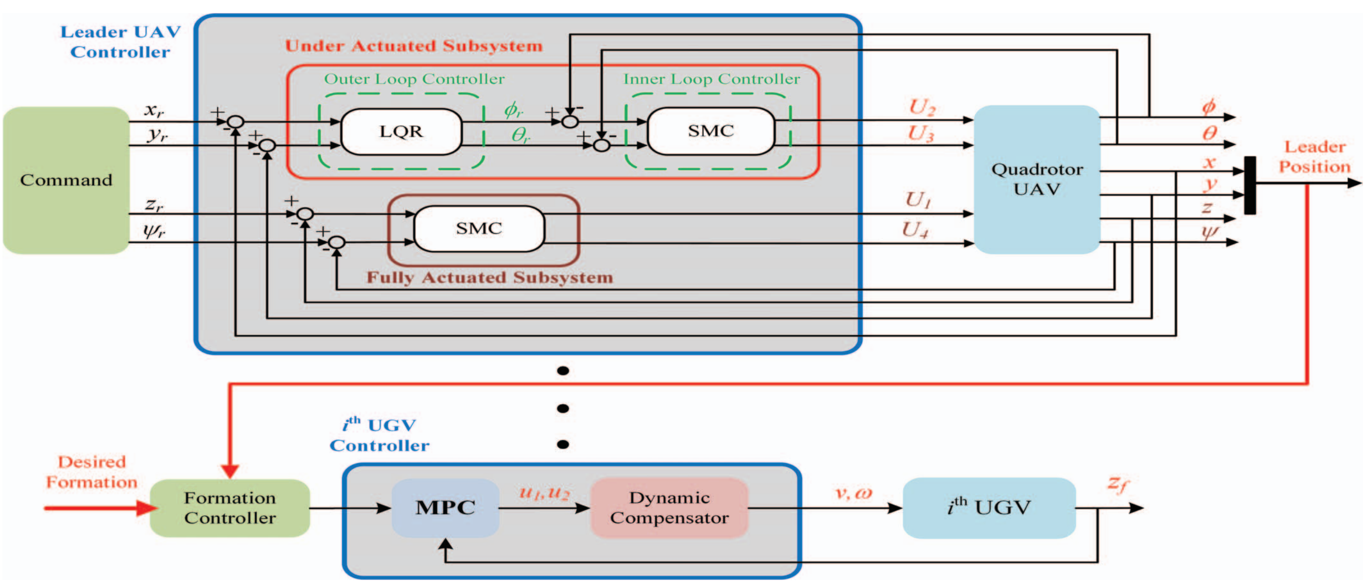

Fig. 2. Overall UAV-UGVs formation control system block diagram

Fig. 3. Task re-assignment mechanism

estimated the value of γ and they are unable to complete the mission. So the remaining healthy UAVs-UGVs start to reconfigure their formation based on the new situation. i.e, each healthy vehicle switches to a new desired formation F^d.

- If $0 < \gamma_j < 0.65$, then one or more UGVs are subject to actuator fault but are still able to complete the mission with a degraded performance. So the full team will continue the mission with a degraded performance.

1) Task Re-Assignment: In case of severe faults, the FTCC problem is solved as an assignment problem. Once the leader UAV receives the fault decision from the FDD unit, the leader UAV knows that the remaining number of UGV followers is $F = N-1$. The leader UAV sends a new formation shape S parametrized by a vector r in Cartesian coordinates relative to the leader position. The desired formation of F followers is assumed as slots to be filled, whilst each follower needs to be assigned to only one of the slots. This can be formulated as an optimal assignment problem, where the cost function $c_{ij} = c(F_i, S_j)$ deciding the cost of assigning the UGV F_i to slot S_j. The cost considered here in this work is to minimize the distance between the UGV follower and the assigned slot. Also the leader UAV has a pre-defined formation shapes according to the number of remaining UGVs. Fig. 4 shows the possible formation shapes for six UAVs-UGVs or fewer.

The optimal assignment problem can be mathematically formulated as follows.

Definition 1: Let $F = \{F_1, F_2, \ldots, F_n\}$ denote the healthy followers UGVs and $S = \{S_1, S_2, \ldots, S_n\}$ denote the slots. Given an $n \times n$ cost matrix where the element at the i^{th} row and the j^{th} column corresponds to the cost of assigning the i^{th} follower UGV to the j^{th} slot, find a permutation π of $\{1, 2, \ldots, n\}$ for which

$$\sum_{i=1}^{n} c(F_i S_{\pi(i)}) \tag{10}$$

is a minimum.

Let $x_{ij} = 1$ denote F_i occupying S_j and 0 otherwise. Then the optimization objective function is represented as:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{11}$$

subject to

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j, \qquad \sum_{j=1}^{n} x_{ij} = 1 \quad \forall i \tag{12}$$

The above constraints presented in Eq. (12) ensure the unique assignment, i.e., one UGV occupies one and only one slot. The Hungarian algorithm can reduce the complexity of finding the optimal assignment from combinatorial to polynomial in time. The input to this algorithm is the $n \times n$ cost matrix which is defined in Definition 1. The solution can be obtained using a bipartite graph where there are n vertices representing the healthy UGV followers, n vertices representing the slots, and edges connecting the followers and slots where each edge has a non-negative cost c_{ij}.

2) Motion Re-Coordination: In the case that the faulty UGV can still complete the mission, the other healthy UAVs-UGVs should reconfigure their controllers within the capability of the faulty UGV. The problem is that the leader UAV already tracks a pre-defined trajectory. So, the pre-defined trajectory should be updated considering this faulty situation.

The idea of motion re-coordination is that: once the leader UAV receives the faulty signal from the faulty UGV and it is able to complete the mission regardless the fault, then the leader UAV will re-generate its desired trajectory states corresponding to the value of γ, i.e. it re-generates the values of x_r, y_r, $\dot{x}_r$, and $\dot{y}_r$ respectively as follows:

$$\begin{bmatrix} \tilde{x}_r & \tilde{y}_r & \tilde{\dot{x}}_r & \tilde{\dot{y}}_r \end{bmatrix}^T = (1-\gamma) \begin{bmatrix} x_r & y_r & \dot{x}_r & \dot{y}_r \end{bmatrix}^T, \tag{13}$$

where $\tilde{x}_r$, $\tilde{y}_r$, $\tilde{\dot{x}}_r$, and $\tilde{\dot{y}}_r$ are the states of the re-planned trajectory.

Following the steps presented in [9], the followers' controllers will ensure the followers to follow the leader considering the new capabilities of the leader UAV and the faulty UGV. Consequently, the performance of the whole team degrades as the capabilities reduce due to the fault. Thus, the mission can still continue but with degraded performance. Therefore, it reduces but with expected formation performance should be achieved in such a case. Such a control strategy is referred as to graceful performance degradation as defined in [10] for single vehicle cases.

B. UGVs' Collision Avoidance Algorithm

During the formation reconfiguration, the healthy UGVs should avoid the collision with the faulty UGVs based on the UGVs' sensory information. The reactive obstacle avoidance algorithm presented is based on the mechanical impedance principle. The main concept is to link the movement of each UGV in formation to a virtual repulsive forces based on its

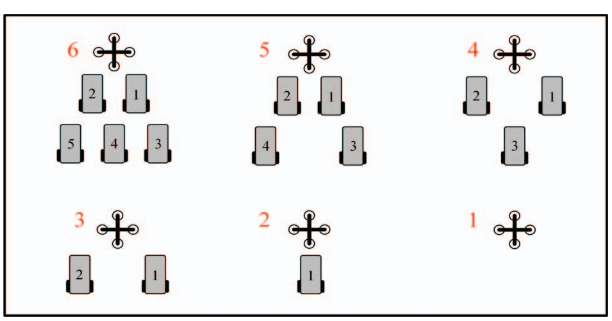

Fig. 4. Possible formation shapes for six or fewer robots

interaction with the surrounding environment. Then the linear and angular velocities of the UGV are changed in response to this repulsive force. More details of the proposed collision avoidance algorithm in the fault-free case can be found in authors' previous work [11].

IV. EXPERIMENTAL RESULTS ANALYSIS

The control strategies discussed in Sections II and III are successfully implemented on a team of UAVs-UGVs. This work is performed in the Networked Autonomous Vehicles Laboratory (NAV Lab) in the Department of Mechanical and Industrial Engineering, Concordia University. All the videos of the experiments performed can be found in the NAV Lab YouTube Channel [12].

The experimental testbed includes the Qball-X4 as the UAV, Quanser QGV as the UGV, the ground station PC, and 24 OptiTrack cameras system as shown in Fig. 5. Both the Qball-X4 and QGV are developed by Quanser Inc. The Qball-X4 is a quadrotor helicopter design, propelled by four motors fitted with 10-inch propellers. The entire quadrotor is enclosed within a protective carbon fiber cage that ensures safe operation of the UAV. Quanser QGV is a two wheeled unicycle mobile robot with a 4 degrees-of-freedom robotic manipulator. Both Qball-X4 and QGV control module is comprised of a data acquisition board (HiQ DAQ) and an embedded Gumstix computer where QuaRC is the Quanser's real-time control software. Together with the Gumstix embedded computer, the HiQ controls the vehicle by reading on-board sensors and sending motor commands. The motor speed controller is connected to two PWM servo outputs on the HiQ. The on-board Gumstix computer runs QuaRC, allowing rapidly develop and deploy controllers for Qball-X4 and QGV real-time control. Runtime sensors measurement, data logging, and parameter tuning are supported between the ground host computer and both the Qball-X4 and the QGV.

Since the experiments taking place indoor in the absence of GPS, then the high level controller implemented on a PC receives the position information of the both the Qball-X4 and the QGV from the vision system consisting of 24 OptiTrack camera system. The high-level controller uses this information to calculate the desired pulse width modulation (PWM) to be sent to the driving motors. The ground station PC used has a processor Intel(R) Core(TM) i7-3770 CPU @ 3.40 GHz, and 4 GB RAM. All the values of the UAV controller parameters are shown in Table I.

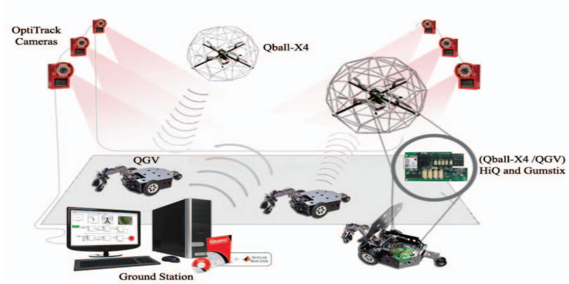

Fig. 5. Experimental setup

Two cases are presented in this work:

- Case 1: A severe fault occurs in the second follower.
- Case 2: A fault occurs in the second follower. However it can still complete the mission.

A. Experimental Scenario

The experiment is performed with one UAV as the leader, and two UGVs as followers. The faulty UGV is considered virtual due to space limitation in the laboratory. In the experiment, the units used are: *m* for position, *deg* for orientation, *m/s* for linear velocity, and *rad/s* for angular velocity. The initial position (x_0, y_0, z_0) of the leader UAV is $[-0.5652, 0.4467, 0]^T$. The initial position of the first UGV follower is $q_1(0) = [-1.1, 0.98, 45°]^T$, and the second UGV follower (the virtual one) is $q_1(0) = [-1.1, 0, 60°]^T$. The objective is to achieve a triangular formation with F_{dl1} = 0.75 m and 120°, F_{dl2} = 0.75 m and 240°.

B. Analysis of Experimental Results of Case 1

In this case, it is assumed that a severe actuator fault occurs at time instant t = 25 sec in the second follower which leads to the mission incompletion. According to this situation, the leader UAV sends a new formation command to the remaining healthy UGV. According to the possible formation presented in Fig. 4, new formation data are sent to the first UGV follower which is 0.75 m and 90°. Fig. 6 illustrates the vehicles' trajectories during mission execution. As can be observed, the team starts the formation in a triangular shape (marked with the • marker), and ends in the form of line formation due to the fault occurrence in the second follower UGV that stops and gets out from the formation. Moreover, Fig. 7 illustrates the formation angle of the first follower. As can be seen, initially the formation angle between the leader UAV and the first follower UGV is 140°, once the UAV reaches the desired height, the UGV tries to achieve the desired ϕ_d which is 120°. When fault occurs, within two seconds of fault occurrence, the first follower UGV starts accommodating the fault effect on the mission and reconfigure the formation to the new shape.

C. Analysis of Experimental Results of Case 2

In this case, fault is injected to the left motor of the second follower at t = 25 sec. The loss of effectiveness of the left motor is about 38%. So, the second UGV follower is still capable of continuing the mission with degraded performance. The faulty follower sends the fault information to the leader UAV. As a result, and according to Eq. (13), the leader UAV updates the reference trajectory states. From Fig. 8, the whole team continues the mission accommodating the fault effect. In Fig. 9, the proposed motion re-coordination

TABLE I

UAV CONTROLLER PARAMETERS

J_{xx} = 0.03 kg.m^2	J_{yy} = 0.03 kg.m^2	M = 1.4 kg
J_{zz} = 0.04 kg.m^2	K_{yaw} =4 N.m	L = 0.2 m
K = 120 N	ω_m =15 rad/sec	

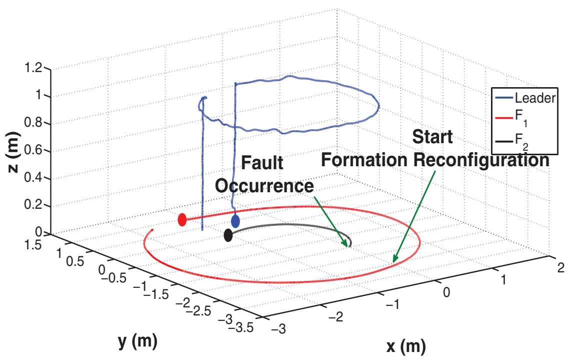

Fig. 6. Robots trajectories during mission execution in Case 1

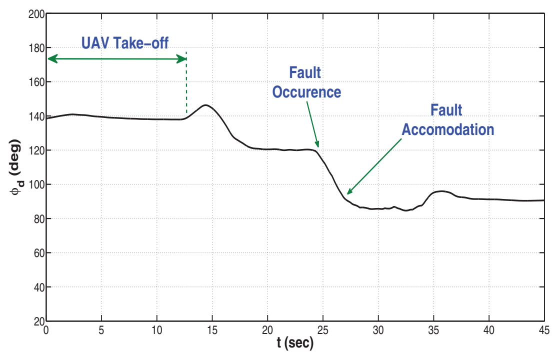

Fig. 7. Formation angle of the first follower in Case 1

is achieved. All the UAV-UGVs after 25 sec reduce their linear velocities by incorporating the fault occurred in the second follower. Each vehicle moves only with about 62% of its capability. If the healthy UAV-UGV do not reduce their capabilities, then the desired formation configuration cannot be maintained due to the degraded capability of the faulty UGV.

V. CONCLUSION

FTCC problem of a team of UAVs-UGVs is presented in this paper. The proposed FTCC scheme is capable of 1) reconfiguring the formation in case that one or more UGVs are faulty and cannot complete the mission; and 2) re-coordinating the motion of each UAV-UGV in the team

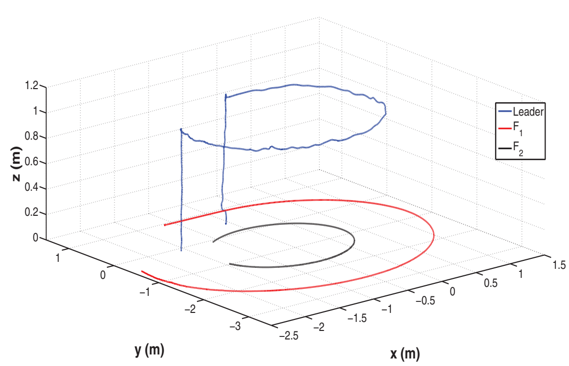

Fig. 8. Robots trajectories during mission execution in Case 2

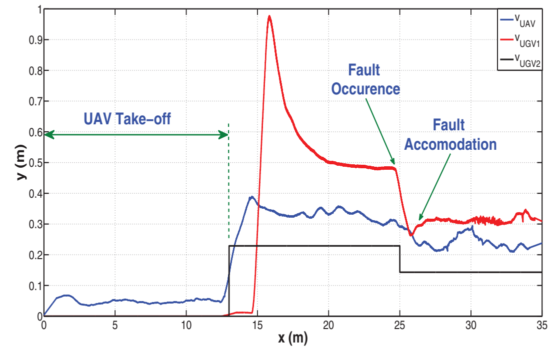

Fig. 9. Robots' velocities in Case 2

if one or more UGVs subject to faults but can still continue the mission with degraded performance. Experimental results illustrate that the formation system is stabilized, converges to the desired formation, reconfigures the formation shape in the presence of severe actuator faults, and re-coordinates the mission in the case of non-severe faults.

REFERENCES

[1] M. A. Kamel, K. A. Ghamry, and Y. M. Zhang, "Fault tolerant cooperative control of multiple UAVs-UGVs under actuator faults," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 644–649.

[2] Y. M. Zhang and H. Mehrjerdi, "A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2013, pp. 1087–1096.

[3] H. A. Izadi, B. W. Gordon, and Y. M. Zhang, "Hierarchical decentralized receding horizon control of multiple vehicles with communication failures," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 744–759, 2013.

[4] F. A. P. Lie and T. H. Go, "Reconfiguration control with collision avoidance framework for unmanned aerial vehicles in three-dimensional space," *Journal of Aerospace Engineering*, vol. 26, no. 3, pp. 637–645, 2011.

[5] A. Chamseddine, Y. M. Zhang, and C. A. Rabbath, "Trajectory planning and re-planning for fault tolerant formation flight control of quadrotor unmanned aerial vehicles," in *American Control Conference (ACC)*, 2012, pp. 3291–3296.

[6] M. A. Kamel, Y. M. Zhang, and X. Yu, "Fault-tolerant cooperative control of multiple wheeled mobile robots under actuator faults," in *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS)*, 2015, pp. 1152–1157.

[7] X. Yu, Z. Liu, and Y. M. Zhang, "Fault-tolerant formation control of multiple UAVs in the presence of actuator faults," *International Journal of Robust and Nonlinear Control*, 2015. [Online]. Available: DOI:10.1002/rnc.3467

[8] K. A. Ghamry and Y. M. Zhang, "Formation control of multiple quadrotors based on leader-follower method," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 1037–1042.

[9] M. A. Kamel and Y. M. Zhang, "Linear model predictive control via feedback linearization for formation control of multiple wheeled mobile robots," in *IEEE International Conference on Information and Automation*, 2015, pp. 1283–1288.

[10] Y. M. Zhang and J. Jiang, "Fault tolerant control system design with explicit consideration of performance degradation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 838–848, 2003.

[11] M. A. Kamel and Y. M. Zhang, "Decentralized leader-follower formation control with obstacle avoidance of multiple unicycle mobile robots," in *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2015, pp. 406–411.

[12] "NAV Laboratroy YouTube Channel," https://www.youtube.com/user/NAVConcordia/videos, 2015.