INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
*Int. J. Adapt. Control Signal Process.* 2015; **29**:1–23
Published online 25 November 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/acs.2451

# Active fault-tolerant control system design with trajectory re-planning against actuator faults and saturation: Application to a quadrotor unmanned aerial vehicle

A. Chamseddine[1], D. Theilliol[2,3], Y. M. Zhang[1,*,†], C. Join[2,3] and C. A. Rabbath[1]

[1]*Department of Mechanical and Industrial Engineering, Concordia University, Montreal, Quebec, Canada H3G 1M8*
[2]*Faculté des Sciences et Techniques, University of Lorraine, Vandoeuvre-lès-Nancy, France*
[3]*CNRS, CRAN, UMR 7039, France*

## SUMMARY

During the past 30 years, various fault-tolerant control (FTC) methods have been developed to address actuator or component faults for various systems with or without tracking control objectives. However, very few FTC strategies establish a relation between the post-fault reference trajectory to track and the remaining resources in the system after fault occurrence. This is an open problem that is not well considered in the literature. The main contribution of this paper is in the design of a reconfigurable FTC and trajectory planning scheme with emphasis on online decision making using differential flatness. In the fault-free case and on the basis of the available actuator resources, the reference trajectories are synthesized so as to drive the system as fast as possible to its desired setpoint without violating system constraints. In the fault case, the proposed active FTC system (AFTCS) consists in synthesizing a reconfigurable feedback control along with a modified reference trajectories once an actuator fault has been diagnosed by a fault detection and diagnosis scheme, which uses a parameter-estimation-based unscented Kalman filter. Benefited with the integration of trajectory re-planning using the flatness concept and the compensation-based reconfigurable controller, both faults and saturation in actuators can be handled effectively with the proposed AFTCS design. Advantages and limitations of the proposed AFTCS are illustrated using an experimental quadrotor unmanned aerial vehicle testbed. Copyright © 2013 John Wiley & Sons, Ltd.

Received 16 April 2012; Revised 24 August 2013; Accepted 25 October 2013

KEY WORDS: actuator faults; fault detection and diagnosis (FDD); reconfiguration mechanism; reconfigurable reference trajectory; active fault-tolerant control system (AFTCS); unmanned aerial vehicle (UAV)

## NOMENCLATURE

| | |
|---|---|
| $\mathcal{F}$ | flat outputs |
| $\mathcal{F}^*$ | reference trajectories for the flat outputs $F$ in the nominal fault-free case |
| $\mathcal{F}^*_{rem}$ | remaining reference trajectories to track after fault occurrence |
| $\mathcal{F}^*_{rep}$ | re-planned reference trajectories in the fault case |
| $u^{nom}$ | nominal control inputs in the nominal fault-free case |
| $u^{*nom}$ | expected value of the nominal control inputs $u^{nom}$ |
| $u^{add}$ | additive control term added to the nominal control law $u^{nom}$ |
| $u^{FTC}$ | control inputs generated by the fault-tolerant controller |
| $u^{*FTC}$ | expected value of $u^{FTC}$ |
| $t_f$ | final time of the mission |
| $t_{fault}$ | time of fault occurrence |

*Correspondence to: Y. M. Zhang, Department of Mechanical and Industrial Engineering, Concordia University, Montreal, Quebec, Canada H3G 1M8.
†E-mail: ymzhang@encs.concordia.ca

Copyright © 2013 John Wiley & Sons, Ltd.

$t_d$ time of fault detection and diagnosis
$t_r$ time of trajectory re-planning
$t_{frep}$ final time of the mission after trajectory re-planning

## 1. INTRODUCTION

Among various motivations for fault-tolerant control (FTC) research and development, it is emphasized in [1] that about 60% of control performance degradation in industrial plants is due to sensor and actuator failures and equipment fouling. The objective of FTC systems is to maintain current performance close to the desired one and preserve stability in the presence of component and/or instrument faults. The area of FTC has attracted much attention during the past 30 years, where many survey papers and books have been published in this area, for example, [2–10]. The FTC methods can be categorized into passive and active approaches. Active FTC approaches are classically characterized by an online fault detection and diagnosis (FDD) process and control reconfiguration mechanism. On the basis of the information provided by the FDD module, control reconfiguration considers the problem of changing the control law or the controller structure by selecting a new set of inputs and outputs in order to take into account the faults occurred. A recent comparative study in [11] studies active and passive FTC systems by examining the similarities and differences between these two approaches from both philosophical and practical points of view. A recent study on passive and active FTC schemes applied to a quadrotor unmanned aerial vehicle based on sliding mode control technique has been carried out and tested experimentally [12]. The investigation shows the advantages of active FTC over the passive FTC scheme. In this current work, active FTC is considered under trajectory tracking framework.

Many FTC strategies have been developed to address actuator or component faults for linear, linear parameter-varying, and nonlinear systems. However, there are still many open issues yet to be resolved. For instance, faults can be accommodated provided that there are sufficient remaining resources in the system. However, very few FTC strategies establish a relation between the post-fault objectives and the remaining resources in the system after fault occurrence. In other words, when trajectory tracking is considered, the tracking capability after faults is rarely investigated, and no solution is given when the pre-fault trajectory cannot be tracked. Degradation in tracking performance occurs, for example, when actuators hit their limits and cannot deliver the actuation inputs desired by the controller. There is more tendency for this case to take place when actuator faults occur and the system resources decrease. Reference management or reference governor has been proposed in the literature for systems with input and/or state constraints. In [13], a command governor based on conceptual tools of predictive control is designed for solving setpoint tracking problems wherein pointwise-in-time input and/or state inequality constraints are present. A reference governor is designed in [14] for general discrete-time and continuous-time nonlinear systems with uncertainties. It relies on safety properties provided by sublevel sets of equilibria-parameterized functions. In the context of FTC, a reference input management is introduced in [15] to determine appropriate reference inputs in the presence of actuator faults to avoid potential saturation. The concept was extended to the case with multiple faults in [16]. An online adjustment strategy of reference input trajectories has been proposed in [17] based on an optimal energy principle. Another reference inputs generation method is proposed in [18] where the objective is to minimize the distance between the desirable output vector before and after failure while distributing the most equitably the energy among the healthy actuators.

With respect to the aforementioned works, the main contribution of this paper resides in the integration of an online reconfigurable controller and reference trajectory re-planning scheme with emphasis on online decision making using differential flatness. The main advantage of differential flatness is that it allows to determine *a priori* the control inputs to be applied along the desired reference trajectories. This makes possible for the reconfiguration scheme to take into consideration the remaining resources in the system after fault occurrence with an objective to increase the performance of the active FTC system (AFTCS) under trajectory tracking scenario. In the fault-free case and on the basis of the available actuator resources, the reference trajectories are synthesized so as to drive the system as fast as possible to its desired setpoint without violating system constraints.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

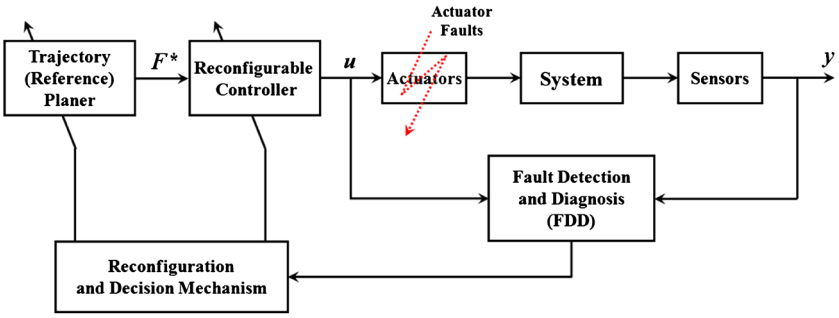


Figure 1. General architecture of the proposed active fault-tolerant control system.

When actuator faults take place, the faults are detected and diagnosed by a parameter-estimation-based FDD scheme using a modified unscented Kalman filter (UKF). The proposed FTC method consists in synthesizing a reconfigurable feedback control along with a modified reference trajectory when performance degradation occurs because of faults and saturation in actuators. The basic concept of the flatness principle and integrated structure of the proposed AFTCS are presented in Section 2. The approach is then applied to linear time-invariant (LTI) systems in Section 3. The effectiveness of the integrated active FTC scheme is illustrated in Section 4 by using a quadrotor unmanned aerial vehicle (UAV) testbed subject to actuator faults. Conclusions and future work are discussed in Section 5.

## 2. GENERAL CONCEPT AND STRUCTURE OF THE PROPOSED SCHEME

A functional block diagram that shows the overall architecture of the proposed AFTCS is illustrated in Figure 1. An FDD module detects, isolates, and identifies faults and provides real-time information for a reconfiguration mechanism to both controller and reference trajectory planer. The reconfiguration mechanism comprises a decision module that decides which block(s) to reconfigure among the controller and the reference trajectory and whether the reconfiguration includes readjustment of the controller gains and/or the structure.

This paper considers a reference trajectory planning/re-planning approach based on differential flatness and integrated with a fault compensation/retrofit FTC scheme and a UKF-based FDD method. In the fault-free case, the reference trajectories are designed so as to drive the system as fast as possible to the setpoint while respecting system constraints. Once a fault is detected, isolated, and identified, the reconfiguration mechanism will reconfigure the reconfigurable controller and re-plan the reference trajectories if needed so as to take into consideration the new system constraints based on information provided by the FDD module. The decision on the reconfiguration must be taken by the decision module based on the occurred actuator faults and the post-fault system constraints. Such decision must be taken in real-time in order to recover as fast as possible the nominal performance or desired reduced/degraded performance gracefully [15–17, 19]. The subsequent section presents the general concept of the proposed approach.

### 2.1. *Trajectory planning and decision making: general concept*

The reconfiguration decision and the reference trajectories planning/re-planning are based on differential flatness. Flatness can be defined as follows where $f^{(k)}$ denotes the $k^{th}$ derivative of function $f$.

*Definition 1*
Let a system model be given by differential equations of the form

$$S_i\left(\xi, \dot{\xi}, \ddot{\xi}, \ldots, \xi^{(\sigma_i)}\right) = 0, \quad i = 1, \ldots, q \tag{1}$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

where $\xi = (\xi_1, \ldots, \xi_{n+m})$ are the system variables including also the input variables ($n$ is the number of state variables, and $m$ is the number of control variables). Such a system is called (differentially) flat [20, 21], if there exists an $m$-tuple $\mathcal{F}$ of functions:

$$\mathcal{F}_j = \psi_j \left(\xi, \dot{\xi}, \ddot{\xi}, \ldots, \xi^{(\eta_j)}\right), \quad j = 1, \ldots, m \tag{2}$$

fulfilling the following conditions:

1. The components of $\mathcal{F}$ are differentially independent; that is, there does not exist a differential equation of the form

$$R\left(\mathcal{F}, \dot{\mathcal{F}}, \ddot{\mathcal{F}}, \ldots, \mathcal{F}^{(\alpha)}\right) = 0 \tag{3}$$

   which means such a relation in $\mathcal{F}$ only cannot be derived from (1).
2. The system variables in $\xi$ can be expressed by functions of $\mathcal{F}$ and its time derivatives:

$$\xi_l = \phi_l \left(\mathcal{F}, \dot{\mathcal{F}}, \ddot{\mathcal{F}}, \ldots, \mathcal{F}^{(\gamma_l)}\right), \quad l = 1, \ldots, n+m \tag{4}$$

   This implies that the trajectories of $\mathcal{F}$ provide a parameterization for the trajectories of the system variables (including the input variables).

*2.1.1. Trajectory planning.* The second condition of Definition 1 is a key property in the trajectory planning and reference input design: if the trajectories of $\mathcal{F}$ are *a priori* known, then one can predict the trajectories that the system variables will follow when the system is forced to track the trajectories of $\mathcal{F}$. To this end, define $\mathcal{F}^*$ as the set of reference trajectories of $\mathcal{F}$. According to (4), forcing the $m$-tuple $\mathcal{F}$ to track the reference trajectories $\mathcal{F}^*$ implies that the expected evolutions of the system variables, denoted as $\xi_l^*$, are as follows:

$$\xi_l^* = \phi_l \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right), \quad l = 1, \ldots, n+m \tag{5}$$

Path or trajectory planning is a complex problem because it involves meeting the physical constraints of the plant, constraints from the operating environment, and other operational requirements [22]. The key objective to be met is that paths must be feasible, where feasible paths are those that meet the kinematic constraints as well as the physical limitations of the plant including actuator constraints. It is assumed in this current work that the operating environment is constraint free, and thus, only operational requirements and plant physical constraints are considered. This leads to the following definition.

*Definition 2*
The domain $\Omega$ can be defined as the set of variables where operational requirements and plant physical constraints are respected:

$$\Omega = \left\{\xi \in \Re^{n+m} : \underline{\xi} \leq \xi \leq \overline{\xi}\right\} \subset \Re^{n+m} \tag{6}$$

where $\underline{\xi}$ and $\overline{\xi}$ are the lower and upper bounds of $\xi$, respectively.

It turns out from (5) and (6) that the reference trajectories $\mathcal{F}^*$ are feasible in the fault-free case if they are chosen such that the trajectories of the system variables remain at the interior of domain $\Omega$:

$$\underline{\xi}_l \leq \phi_l \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right) \leq \overline{\xi}_l \quad , \quad l = 1, \ldots, n+m \tag{7}$$

The reference trajectories $\mathcal{F}^*$ are determined as function of the initial conditions of the system and the final desired conditions as well as the initial time $t_0$ and the final time $t_f$. The initial and final conditions and the initial time $t_0$ are known, and the sole unknown is the final time $t_f$. Generally speaking, it is required to attain the desired objectives as fast as possible while respecting system

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

constraints. The trajectory planning problem can then be formulated as an optimization problem as follows:

$$P \begin{cases} \text{Minimize} & t_f - t_0 \\ \text{Subject to} & \underline{\xi}_l \leqslant \phi_l \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right) \leqslant \overline{\xi}_l \quad , \quad l = 1, \ldots, n+m \end{cases} \tag{8}$$

*2.1.2. Reconfiguration mechanism and decision making.* Once a fault occurs at time instant $t_{fault}$, the domain $\Omega$ will change depending on the nature and amplitude of the fault. Similarly as for domain $\Omega$, the domain $\Omega^f$ in the fault case can be defined as follows.

*Definition 3*
Upon actuator faults, the operational requirements and the physical constraints will change, resulting in a domain $\Omega^f$, which is not equivalent to $\Omega$. The domain $\Omega^f$ can then be defined as follows:

$$\Omega^f = \left\{\xi^f \in \Re^{n+m} : \underline{\xi}^f \leqslant \xi^f \leqslant \overline{\xi}^f\right\} \subset \Re^{n+m} \tag{9}$$

where $\xi^f$ is employed to denote the plant variables after fault occurrence. $\underline{\xi}^f$ and $\overline{\xi}^f$ are the post-fault lower and upper bounds of $\xi^f$, respectively.

It can also be noted that the system dynamics will change, and instead of the parameterization in (4), the system variables in $\xi^f$ expressed by functions of $\mathcal{F}$ and its time derivatives are as follows:

$$\xi_l^f = \phi_l^f \left(\mathcal{F}, \dot{\mathcal{F}}, \ddot{\mathcal{F}}, \ldots, \mathcal{F}^{(\gamma_l)}\right), \quad l = 1, \ldots, n+m \tag{10}$$

After fault occurrence, an FTC is applied to the system to compensate the occurred fault. The controller together with the FTC module will continue to force $\mathcal{F}$ to track their reference trajectories $\mathcal{F}^*$. This implies that the expected trajectories of the system variables in the fault case will be the following:

$$\xi_l^{*f} = \phi_l^f \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right), \quad l = 1, \ldots, n+m \tag{11}$$

The reference trajectories $\mathcal{F}^*$ are designed so that the operational and physical constraints expressed by (7) are verified. However, there is no guarantee that the post-fault constraints will be respected when the system is forced to track the same pre-fault trajectories. In other words, there is no guarantee that the following equations will be verified:

$$\underline{\xi}^f \leqslant \phi_l^f \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right) \leqslant \overline{\xi}^f \quad , \quad l = 1, \ldots, n+m \tag{12}$$

Therefore, the pre-fault trajectories may need to be changed depending on the system post-fault conditions. The values of the post-fault constraints $\underline{\xi}^f$ and $\overline{\xi}^f$ will be available at time instant $t_d$ by an FDD module. The decision on whether to re-plan or not the pre-fault trajectories depends on the system conditions at fault instant as well as on the fault location and amplitude. Denote $\mathcal{F}_{rem}^*$ as the remaining reference trajectories to track after fault occurrence. The pre-fault trajectory must be re-planned only if the constraints are not respected for a part of the remaining trajectories $\mathcal{F}_{rem}^*$; that is, when the following equations are not verified:

$$\underline{\xi}^f \leqslant \phi_l^f \left(\mathcal{F}_{rem}^*, \dot{\mathcal{F}}_{rem}^*, \ddot{\mathcal{F}}_{rem}^*, \ldots, \mathcal{F}_{rem}^{*(\gamma_l)}\right) \leqslant \overline{\xi}^f \quad , \quad l = 1, \ldots, n+m \tag{13}$$

In this case, the pre-fault trajectories are re-planned at time instant $t_r$ where $\mathcal{F}_{rep}^*$ denote the re-planned reference trajectories in the fault condition. $\mathcal{F}_{rep}^*$ must be determined so that the post-fault constraints $\underline{\xi}^f$ and $\overline{\xi}^f$ are respected; that is, the following relations are verified:

$$\underline{\xi}^f \leqslant \phi_l^f \left(\mathcal{F}_{rep}^*, \dot{\mathcal{F}}_{rep}^*, \ddot{\mathcal{F}}_{rep}^*, \ldots, \mathcal{F}_{rep}^{*(\gamma_l)}\right) \leqslant \overline{\xi}^f \quad , \quad l = 1, \ldots, n+m \tag{14}$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

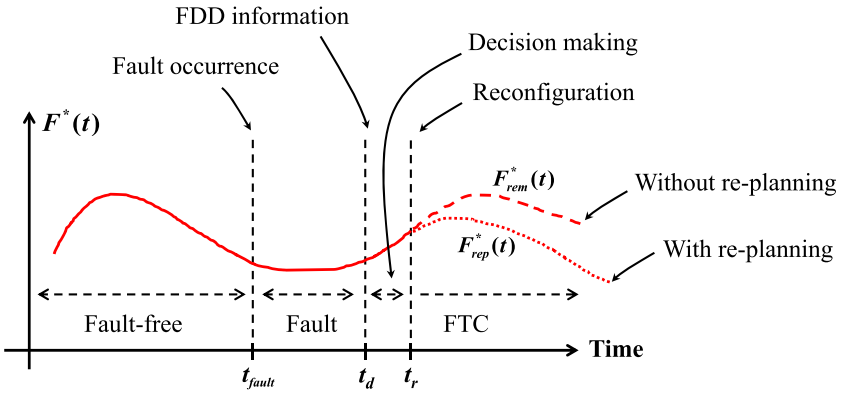


Figure 2. Principal idea of trajectory re-planning. FDD, fault detection and diagnosis; FTC, fault-tolerant control.

The trajectory re-planning in the fault case can also be posed as an optimization problem as follows:

$$P^f \begin{cases} \text{Minimize} & t_{f_{rep}} - t_r \\ \text{Subject to} & \underline{\xi}^f \leqslant \phi_l^f \left( \mathcal{F}^*_{rep}, \dot{\mathcal{F}}^*_{rep}, \ddot{\mathcal{F}}^*_{rep}, \ldots, \mathcal{F}^{*(\gamma_l)}_{rep} \right) \leqslant \overline{\xi}^f \quad , \quad l = 1, \ldots, n+m \end{cases} \tag{15}$$

where $t_{f_{rep}}$ is the final time for the re-planned trajectory and $t_r$ is the instant of re-planning. Figure 2 shows the principal idea of trajectory re-planning in the case of fault occurrence: when a fault takes place at time instant $t_{fault}$, the FDD module will need some time to detect and diagnose the fault. Once the fault is detected and diagnosed at time instant $t_d$, the post-fault constraints $\underline{\xi}^f$ and $\overline{\xi}^f$ will be available. On the basis of this information, the decision module must decide whether the trajectory must be kept the same as $\mathcal{F}^*_{rem}$ (dashed line in Figure 2) or re-planned as $\mathcal{F}^*_{rep}$ (dotted line in Figure 2). The trajectory re-planning takes place at time instant $t_r$. It turns out that an FDD module is an important part of the whole scheme because it is needed to provide the necessary information on the occurred fault, as shown in Figure 1 with the proposed overall FTC system configuration. Such information helps in deciding whether or not to re-plan the pre-fault trajectories and also helps in determining the re-planned ones $\mathcal{F}^*_{rep}$. Together with the trajectory re-planning, the FTC is applied at time instant $t_r$ to compensate the fault effect and to recover as fast as possible the pre-fault or gracefully degraded performance.

## 3. LINEAR TIME-INVARIANT SYSTEMS DESIGN

The general ideas proposed in the previous section can be applied to LTI systems. To this end, consider the following system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{16}$$

where $A \in \Re^{n \times n}$, $B \in \Re^{n \times m}$ and $C \in \Re^{p \times n}$ are respectively the state, control, and output matrices. $x(t) \in \Re^n$ is the system state vector, $u(t) \in \Re^m$ is the control input vector, and $y(t) \in \Re^p$ is the system output vector. Linear systems are differentially flat, and the components of the $m$-tuple $\mathcal{F}(t)$ are subset of the system state variables $x(t)$:

$$\mathcal{F}(t) = C_f x(t) \tag{17}$$

where $C_f \in \Re^{m \times n}$. Define $\mathcal{F}_i(t)$ $(i = 1, \ldots, m)$ as the $i^{th}$ component of the $m$-tuple $\mathcal{F}(t)$. In the sequel, $\mathcal{F}_i(t)$ will be referred to as flat outputs. The parameterization of the control inputs $u(t)$ as

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

function of $\mathcal{F}_i(t)$ can be achieved by taking each flat output:

$$\mathcal{F}_i(t) = C_f^i x(t) \tag{18}$$

where $C_f^i$ is the $i^{th}$ row of matrix $C_f$ and differentiate with respect to time until at least one of the control inputs appear explicitly. This is the definition of the well-known relative degree as presented in [23], and such a parameterization of the control inputs $u(t)$ as function of $\mathcal{F}_i(t)$ is the key point/concept for linking flat outputs with desired control inputs through the use of flatness theory so that the planned trajectory can naturally respect to system constraints for trajectory planning/re-planning.

*Definition 4*
The linear system $(A, B, C_f)$ has vector relative degree $r = (r_1, \ldots, r_m) \in \mathcal{N}^{1\times m}$ if and only if

(a) $\forall i \in \{1, \ldots, m\}, \forall k \in \{0, \ldots, r_i - 2\} : C_f^i A^k B = 0_{1\times m}$

(b) $\text{rank} \begin{bmatrix} C_f^1 A^{r_1-1} B \\ C_f^2 A^{r_2-1} B \\ \vdots \\ C_f^m A^{r_m-1} B \end{bmatrix} = m$

It is straightforward to show that when differentiating the $i^{th}$ flat output $r_i$ times, one obtains the following:

$$\mathcal{F}_i^{(r_i)}(t) = C_f^i A^{r_i} x(t) + C_f^i A^{r_i-1} B u(t) \quad ; \quad i = 1, \ldots, m \tag{19}$$

where $C_f^i A^{r_i-1} B \neq 0$ according to the aforementioned definition. In matrix form

$$\begin{pmatrix} \mathcal{F}_1^{(r_1)}(t) \\ \mathcal{F}_2^{(r_2)}(t) \\ \vdots \\ \mathcal{F}_m^{(r_m)}(t) \end{pmatrix} = \begin{pmatrix} C_f^1 A^{r_1} \\ C_f^2 A^{r_2} \\ \vdots \\ C_f^m A^{r_m} \end{pmatrix} C_f^+ \mathcal{F}(t) + \begin{pmatrix} C_f^1 A^{r_1-1} B \\ C_f^2 A^{r_2-1} B \\ \vdots \\ C_f^m A^{r_m-1} B \end{pmatrix} u(t) \tag{20}$$

where $C_f^+$ is the pseudo-inverse of $C_f$. Finally, the control inputs $u(t)$ can be parameterized as function of the flat outputs and their derivatives as follows:

$$u(t) = \begin{pmatrix} C_f^1 A^{r_1-1} B \\ C_f^2 A^{r_2-1} B \\ \vdots \\ C_f^m A^{r_m-1} B \end{pmatrix}^{-1} \left[ \begin{pmatrix} \mathcal{F}_1^{(r_1)}(t) \\ \mathcal{F}_2^{(r_2)}(t) \\ \vdots \\ \mathcal{F}_m^{(r_m)}(t) \end{pmatrix} - \begin{pmatrix} C_f^1 A^{r_1} \\ C_f^2 A^{r_2} \\ \vdots \\ C_f^m A^{r_m} \end{pmatrix} C_f^+ \mathcal{F}(t) \right] \tag{21}$$

### *3.1. Nominal fault-free case*

Define $\mathcal{F}^*(t)$ as the reference trajectories of the $m$-tuple $\mathcal{F}(t)$. In the nominal fault-free case, the control objective is to cause $\mathcal{F}(t)$ to track their reference trajectories $\mathcal{F}^*(t)$ in the sense that $\mathcal{F}(t) = \mathcal{F}^*(t)$. The eigenstructure assignment or the linear-quadratic regulator (LQR) is among the most popular controller design techniques for MIMO linear systems [24, 25]. The subsequent development is independent of the employed control technique, and thus, the nominal controller is denoted as $u^{nom}(t)$, where the explicit form of $u^{nom}(t)$ depends on the dynamics of the considered system and the employed control technique. Thanks to flatness, it is possible to determine *a priori* the expected values of the control inputs to be applied to the system in the nominal fault-free case.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

When the system is forced to track the reference trajectories $\mathcal{F}^*(t)$, it is possible to see from (21) that the expected values of the control inputs are as follows:

$$u^{*nom}(t) = \begin{pmatrix} C_f^1 A^{r_1-1} B \\ C_f^2 A^{r_2-1} B \\ \vdots \\ C_f^m A^{r_m-1} B \end{pmatrix}^{-1} \left[ \begin{pmatrix} \mathcal{F}_1^{*(r_1)}(t) \\ \mathcal{F}_2^{*(r_2)}(t) \\ \vdots \\ \mathcal{F}_m^{*(r_m)}(t) \end{pmatrix} - \begin{pmatrix} C_f^1 A^{r_1} \\ C_f^2 A^{r_2} \\ \vdots \\ C_f^m A^{r_m} \end{pmatrix} C_f^+ \mathcal{F}^*(t) \right] \tag{22}$$

In the sequel and for simplification purposes, the aforementioned equation will be denoted as follows:

$$u^{*nom}(t) = \Lambda_1^{-1} \left( \Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t) \right) \tag{23}$$

When actuator constraints are considered, the path planning problem consists in determining the final time $t_f$ so that $\underline{u} \leqslant u^{*nom}(t) \leqslant \overline{u}$ or according to (8):

$$P \begin{cases} \text{Minimize} & t_f - t_0 \\ \text{Subject to} & \underline{u} \leqslant \Lambda_1^{-1} \left( \Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t) \right) \leqslant \overline{u} \end{cases} \tag{24}$$

where $\underline{u}$ and $\overline{u}$ are the lower and upper limits of the control inputs. In most conventional control systems, controllers are designed for fault-free systems without taking into account the possibility of fault occurrence. Therefore, when degradation and damage occur on the actuators, the control inputs $u(t)$ applied to the system are under unexpected and unknown changes. In this study, the considered faults are loss of control effectiveness. For such faults, system (16) can be written as [24, 25]:

$$\begin{aligned} \dot{x}^f(t) &= Ax^f(t) + B\Gamma u^f(t) \\ &= Ax^f(t) + B(I_{m\times m} - W)u^f(t) \end{aligned} \tag{25}$$

where $I_{m\times m}$ is the identity matrix of dimension $m \times m$, and $W$ is a matrix representing the loss of control effectiveness due to actuator faults, and it is defined as $W = diag([w_1, w_2, \ldots, w_m])$. The $i^{th}$ diagonal element $w_i = 1$ if the $i^{th}$ actuator is out of order and $w_i = 0$ if the actuator is in healthy condition. In this paper, the complete loss of an actuator is not considered, and only a partial loss of control effectiveness is investigated, that is, $0 \leqslant w_i < 1$. The matrix $\Gamma$ is defined as $\Gamma = diag([\gamma_1, \gamma_2, \ldots, \gamma_m])$, where $\gamma_i = 1 - w_i$ for $i = 1, \ldots, m$. The trajectory planning problem is investigated in the subsequent sections for two cases: passive and active FTC.

### *3.2. Passive fault-tolerant control*

For comparison purpose with the active FTC to be detailed later, consider in the first place the case when a passive FTC is applied to the system. This can be, for instance, a robust controller that is able to accommodate a predefined set of actuator faults. Continuing to force the system variables $x^f(t)$ to track the reference trajectories, it can be shown that the expected control inputs to be applied in the fault case are as follows:

$$u^{*f}(t) = \Gamma^{-1}\Lambda_1^{-1} \left( \Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t) \right) \tag{26}$$

The reference trajectories $\mathcal{F}^*(t)$ are defined such that (24) is verified. By referring to (24) and (26), one can see that

$$\Gamma^{-1}\underline{u} \leqslant \Gamma^{-1}\Lambda_1^{-1} \left( \Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t) \right) \leqslant \Gamma^{-1}\overline{u} \tag{27}$$

When complete loss of actuators are not considered (i.e., $w_i < 1$ or $\gamma_i > 0$), $\Gamma^{-1}$ is invertible and $\Gamma^{-1}\underline{u} \geqslant \underline{u}$ and $\Gamma^{-1}\overline{u} \geqslant \overline{u}$. Thus, for the remaining trajectories $\mathcal{F}_{rem}^*(t)$ to travel, there is no

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

guarantee that the control inputs in the fault condition satisfy the actuator constraints; that is, the following equation is not necessarily verified:

$$\underline{u} \leqslant \Gamma^{-1}\Lambda_1^{-1}\left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \leqslant \overline{u} \tag{28}$$

If this is the case, the pre-fault reference trajectories $\mathcal{F}^*(t)$ must be redesigned as $\mathcal{F}^*_{rep}(t)$ to verify (28). This can be achieved by calculating the final time $t_{f_{rep}}$ for the re-planned trajectory according to (15):

$$P^f \begin{cases} \text{Minimize} & t_{f_{rep}} - t_r \\ \text{Subject to} & \underline{u} \leqslant \Gamma^{-1}\Lambda_1^{-1}\left(\Lambda_{2,rep} - \Lambda_3 C_f^+ \mathcal{F}^*_{rep}(t)\right) \leqslant \overline{u} \end{cases} \tag{29}$$

or

$$P^f \begin{cases} \text{Minimize} & t_{f_{rep}} - t_r \\ \text{Subject to} & \Gamma\,\underline{u} \leqslant \Lambda_1^{-1}\left(\Lambda_{2,rep} - \Lambda_3 C_f^+ \mathcal{F}^*_{rep}(t)\right) \leqslant \Gamma\,\overline{u} \end{cases} \tag{30}$$

where $\mathcal{F}^*_{rep}(t)$ denotes the re-planned reference trajectories in the fault case and $\Lambda_{2,rep} = \left(\mathcal{F}^{(r_1)}_{1,rep}(t)\ \ \mathcal{F}^{(r_2)}_{2,rep}(t)\ \ \cdots\ \ \mathcal{F}^{(r_m)}_{m,rep}(t)\right)^T$. $t_r$ is the instant of re-planning. It should be noted that the aforementioned development requires the matrix $\Gamma$ to be known, and thus, an FDD module is required to provide this information. The estimation of $\Gamma$ may be uncertain in practice. However, such uncertainty can be easily considered as will be explained in Section 3.4.

### *3.3. Active fault-tolerant control*

Consider now the case when active FTC is applied to the system. In the presence of actuator faults, the faulty actuators corrupt the closed-loop behavior. Moreover, the controller aims at canceling the error between the measurement and its reference input based on fault-free conditions. In this case, the controller gain is away from the optimal one and may drive the system to its physical limitations or even to instability. To ensure that all system variables are within the safe operating range and that all of the control inputs are free from saturation in the event of failures, one has to make appropriate adjustments to the level of control commands as well. The FTC design problem is to determine a controller such that after fault diagnosis, the reconfigured closed-loop system is still stable and that the $m$-tuple $\mathcal{F}(t)$ (which are the system states $x(t)$) track their reference trajectories $\mathcal{F}^*(t)$ with an acceptable level of tracking performance. Under the assumption that an efficient fault diagnosis module is integrated in the reconfigurable control to provide sufficient information about occurred fault, various methods have been proposed to recover as much as possible the performance of the pre-fault system according to the considered fault representation [3]. Few active FTC methods have been designed in order to preserve the output dynamic properties to annihilate the actuator fault effect. For the additive fault representation $B(I_{m\times m} - W)$ defined in (25), the fault compensation principle proposed in [26–28] can be borrowed here for synthesizing a new additive control law $u^{add}(t)$ to be added to the nominal control law $u^{nom}(t)$. Such a fault compensation principle has been also defined in a similar way as retrofit FTC [29]. In this case, the total control signal to be applied to the system is the following:

$$u^{FTC}(t) = u^{nom}(t) + u^{add}(t) \tag{31}$$

When plugging (31) in (25), one obtains

$$\begin{aligned} \dot{x}^f(t) &= Ax^f(t) + B(I_{m\times m} - W)u^{FTC}(t) \\ &= Ax^f(t) + B(I_{m\times m} - W)(u^{nom}(t) + u^{add}(t)) \\ &= Ax^f(t) + Bu^{nom}(t) - BWu^{nom}(t) + B(I_{m\times m} - W)u^{add}(t) \end{aligned} \tag{32}$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

The additional control term $u^{add}(t)$ must be chosen so that (32) and (16) are equivalent. This means that the term $-BWu^{nom}(t) + B(I_{m\times m} - W)u^{add}(t)$ must be zero, and thus, $u^{add}(t)$ can be calculated as follows:

$$u^{add}(t) = [B(I_{m\times m} - W)]^{-1} BWu^{nom}(t) \tag{33}$$

Finally, according to (31), the total control input is the following:

$$\begin{aligned} u^{FTC}(t) &= u^{nom}(t) + [B(I_{m\times m} - W)]^{-1} BWu^{nom}(t) \\ &= \left[I_{m\times m} + [B(I_{m\times m} - W)]^{-1} BW\right] u^{nom}(t) \end{aligned} \tag{34}$$

With the total control input described as (34), the system dynamics in the fault case are the same as in the fault-free case (16), and thus, the control inputs applied to the system are the same as given in (22). However, the total expected control inputs as given in (34) are as follows:

$$\begin{aligned} u^{*FTC}(t) &= \left[I_{m\times m} + [B(I_{m\times m} - W)]^{-1} BW\right] u^{*nom}(t) \\ &= \left[I_{m\times m} + [B(I_{m\times m} - W)]^{-1} BW\right] \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \\ &= \left[I_{m\times m} + (I_{m\times m} - W)^{-1} B^{-1} BW\right] \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \\ &= \left[I_{m\times m} + (I_{m\times m} - W)^{-1} W\right] \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \\ &= \left[I_{m\times m} + \Gamma^{-1}(I_{m\times m} - \Gamma)\right] \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \\ &= \left[I_{m\times m} + \Gamma^{-1} - \Gamma^{-1}\Gamma\right] \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \\ &= \Gamma^{-1} \Lambda_1^{-1} \left(\Lambda_2 - \Lambda_3 C_f^+ \mathcal{F}^*(t)\right) \end{aligned} \tag{35}$$

where $B^{-1}B = I_{m\times m}$ and $\Gamma = I_{m\times m} - W$ are employed to derive the aforementioned equation. The obtained result in (35) is the same as in (26), and this shows that the control inputs to be generated in the fault case remain the same when the system is forced to track the reference trajectories. As a consequence, FTC (whether it is passive or active) does not replace trajectory planning. Re-planning the reference trajectories allows a better management of the post-fault situation in function of the remaining resources in the system. The general concepts presented earlier are applied to a quadrotor UAV testbed in Section 4.

### *3.4. Summary and discussion remarks*

The proposed approach can be summarized as follows:

1. Parameterize the flat outputs as function of system states as expressed in (17).
2. Parameterize the control inputs $u(t)$ as function of the flat outputs and their derivatives as shown in (21).
3. Parameterize the desired trajectories $\mathcal{F}^*(t)$ for the flat outputs as polynomial functions and determine the expected values of the control inputs as in (23).
4. Solve the optimization problem (24) to determine the time of the mission $t_f - t_0$ while satisfying actuator constraints.
5. If actuator faults occur, an FDD module diagnoses the faults and estimates the matrix $\Gamma$.
6. Verify if the expected control inputs violate the actuator constraints for the remaining time of the mission by checking condition (28). If this is the case, then the pre-fault desired trajectories must be re-planned. If not, the mission continues without re-planning.
7. Trajectory re-planning can be achieved by solving the optimization problem (29) to calculate the final time $t_{f_{rep}}$ for the re-planned trajectories.

The parameterization of the desired trajectories as polynomial functions in step 3 as well as solving the optimization problems in steps 4 and 7 will be deeply discussed in the next section.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

The following remarks should be noted:

- The trajectory planning problem (8) is derived using the nominal differential equations (1) without consideration of model uncertainties. This point can be dealt with by introducing a parameter $0 < \rho_\xi < 1$ into the constraints as follows: $\rho_\xi \underline{\xi}_l \leqslant \phi_l \left(\mathcal{F}^*, \dot{\mathcal{F}}^*, \ddot{\mathcal{F}}^*, \ldots, \mathcal{F}^{*(\gamma_l)}\right) \leqslant \rho_\xi \bar{\xi}_l$. This creates a safety margin and robustify the obtained solution $t_f$ against model uncertainties. The value of $\rho_\xi$ can be chosen by trial and error or on the basis of designer's expertise. In the rest of the paper, $\rho_\xi$ will be omitted for the ease of presentation.
- The closed-loop stability is ensured by the control law $u^{nom}(t)$ with a proper choice of structure and gains. Equation (21) is not the control law but represents the parameterization of the control inputs as function of the $m$-tuple $\mathcal{F}$. When the system is forced to track the reference trajectories $\mathcal{F}^*(t)$ (i.e., when $\mathcal{F}(t) = \mathcal{F}^*(t)$), one obtains in (22) the expected control inputs that will be applied to the system along $\mathcal{F}^*(t)$. Even though $u^{*nom}(t)$ and (22) are structurally different and represent two different things, $u^{*nom}(t)$ in (22) gives the expected values of the control inputs that $u^{nom}(t)$ will generate along the reference trajectories $\mathcal{F}^*(t)$, and therefore, it is possible to say that $u^{nom}(t) \approx u^{*nom}(t)$.
- Some FTC approaches are based on model matching principle where the controller gains/structure are completely re-synthesized online [25, 30–32] to guarantee both performance and stability of the post-fault system. The total control inputs to be applied to the system is the same whether the FTC module employs the control law based on the additive control term in (31) or the one based on the intrinsic fault compensation in [25, 30–32]. However, when using the FTC law based on the additive term as in (31), it is possible to express in a straightforward manner as developed in (35) the expected control inputs as function of system dynamics, fault amplitude, and the reference trajectories to track. Another advantage of the control law in (31) is that the synthesized control signal $u^{nom}(t)$ for nominal/fault-free case will not be affected after fault occurrence; only an additive control signal $u^{add}(t)$ in (33) needs to be calculated to compensate the effects of the fault for system reconfiguration. Such a structure 'permits the use of existing test and verification tools making the V&V (verification and validation) procedure substantially simpler than controller redesign', as pointed out in [29].

## 4. APPLICATION TO A QUADROTOR UNMANNED AERIAL VEHICLE SYSTEM

The quadrotor UAV available at the Networked Autonomous Vehicles Lab in the Department of Mechanical and Industrial Engineering of Concordia University is the Qball-X4 testbed (Figure 3),

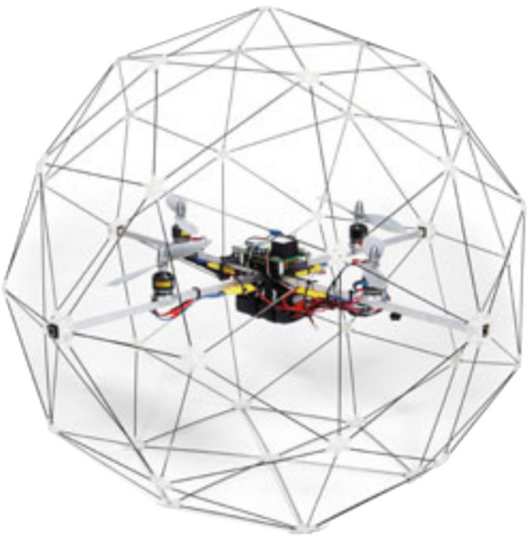

Figure 3. The Qball-X4 unmanned aerial vehicle.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

which was developed by Quanser Inc. partially under the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC) in association with an NSERC Strategic Project Grant led by Concordia University since 2007.

### *4.1. Quadrotor unmanned aerial vehicle description and modeling*

The quadrotor UAV is equipped with the Quanser embedded control module, which is composed of a Quanser HiQ aero data acquisition card and a QuaRC-powered single-board Gumstix embedded computer, where QuaRC is the Quanser's real-time control software. The Quanser HiQ provides high-resolution accelerometer, gyroscope, and magnetometer inertial measurement unit sensors as well as servo outputs to drive four motors. The on-board Gumstix computer runs QuaRC, which allows to rapidly develop and deploy controllers designed in MATLAB/Simulink environment to real-time control the Qball-X4. The controllers run on-board the vehicle itself with a sample time of $T_s = 0.005$ s. Runtime sensors measurement, data logging, and parameter tuning are supported between the ground host computer and the target vehicle. The experiments are taking place indoor in the absence of GPS signals, and thus, the OptiTrack camera system from NaturalPoint is employed to provide the system position in the 3D space. The controller uses this information about the system position as well as the measurements from the on-board sensors to calculate the four pulse-width modulation (PWM) inputs to be sent to each motor to drive the associated propeller of the Qball-X4. A commonly employed quadrotor UAV model [33] is

$$\begin{aligned} m\ddot{x} &= u_z\,(\cos\phi\ \sin\theta\ \cos\psi + \sin\phi\ \sin\psi)\,; \quad & J_1\ddot{\theta} &= u_\theta \\ m\ddot{y} &= u_z\,(\cos\phi\ \sin\theta\ \sin\psi - \sin\phi\ \cos\psi)\,; \quad & J_2\ddot{\phi} &= u_\phi \\ m\ddot{z} &= u_z\,(\cos\phi\ \cos\theta) - mg; & J_3\ddot{\psi} &= u_\psi \end{aligned} \tag{36}$$

where $x$, $y$, and $z$ are the coordinates of the quadrotor UAV center of mass in the earth-fixed frame. $\theta$, $\phi$, and $\psi$ are the pitch, roll, and yaw Euler angles, respectively. $u_z$ is the total lift generated by the four propellers and applied to the quadrotor UAV in the $z$-direction (body-fixed frame). $u_\theta$, $u_\phi$, and $u_\psi$ are respectively the applied torques in $\theta$, $\phi$, and $\psi$ directions. $m$ is the mass, and $J_i$ $(i = 1, 2, 3)$ are the moments of inertia along $y$, $x$, and $z$ directions, respectively. The relations between the lift/torques and the PWM inputs are as follows:

$$\begin{aligned} u_z &= K(u_1 + u_2 + u_3 + u_4) \\ u_\theta &= KL(u_1 - u_2) \\ u_\phi &= KL(u_3 - u_4) \\ u_\psi &= KK_\psi(u_1 + u_2 - u_3 - u_4) \end{aligned} \tag{37}$$

where $u_i$ $(i = 1, \ldots, 4)$ are the PWM inputs, $K$ and $K_\psi$ are positive constants, and $L$ is the distance from the motor to the quadrotor's center of gravity.

### *4.2. Simplified quadrotor model and flat outputs selection*

On the basis of hovering conditions ($u_z \approx mg$ in the $x$ and $y$ directions) with no yawing ($\psi = 0$) and small roll and pitch angles, a simplified linear model can be obtained from (36) as follows:

$$\begin{aligned} \ddot{x}(t) &= \theta(t)g; & J_1\ddot{\theta}(t) &= u_\theta(t) \\ \ddot{y}(t) &= -\phi(t)g; & J_2\ddot{\phi}(t) &= u_\phi(t) \\ \ddot{z}(t) &= u_z(t)/m - g; & J_3\ddot{\psi}(t) &= u_\psi(t) \end{aligned} \tag{38}$$

This simplified model will be used for trajectory planning as well as for the design of the baseline controller. The controller is synthesized on the basis of the LTI model independently to trajectory planning design with more emphasis on tracking in optimal linear-quadratic control. An optimal control law is set up according to the classical stable and robust LQR with integrator technique [34]. Conventional LQR method results in stabilization control only, and thus, an integral action is added

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

to the controller's structure for tracking control. In a state-space representation, the aforementioned model can be written as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_g g \\ y(t) &= Cx(t) \end{aligned} \tag{39}$$

where the matrices $A$, $B$, and $B_g$ are given as

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}; B = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1/m & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1/J_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/J_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/J_3 \end{pmatrix}; B_g = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

and $x(t)$ and $u(t)$ vectors are

$$x^T = \begin{pmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} & \theta & \dot{\theta} & \phi & \dot{\phi} & \psi & \dot{\psi} \end{pmatrix} \text{ and } u^T = \begin{pmatrix} u_z & u_\theta & u_\phi & u_\psi \end{pmatrix}$$

According to the definition of flatness in Section 2, it can be seen that model (38) is flat with choice of the flat outputs $\mathcal{F}_1(t) = z(t)$, $\mathcal{F}_2(t) = x(t)$, $\mathcal{F}_3(t) = y(t)$, and $\mathcal{F}_4(t) = \psi(t)$. The $m$-tuple $\mathcal{F}$ is a subset of the state vector $x$, and therefore, $\mathcal{F}(t) = C_f x(t)$ with

$$C_f = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{40}$$

The parameterization of system's state variables as function of flat outputs is then

$$\begin{aligned} &x(t) = \mathcal{F}_2(t) \;\; ; \;\; y(t) = \mathcal{F}_3(t) \;\; ; \;\; z(t) = \mathcal{F}_1(t) \;\; ; \\ &\theta(t) = \frac{\ddot{\mathcal{F}}_2(t)}{g} \;\; ; \;\; \phi(t) = -\frac{\ddot{\mathcal{F}}_3(t)}{g} \;\; ; \;\; \psi(t) = \mathcal{F}_4(t) \end{aligned} \tag{41}$$

By applying the method presented in Section 3, it is possible to express the flat outputs in function of the control inputs:

$$\begin{pmatrix} \ddot{F}_1(t) \\ F_2^{(4)}(t) \\ F_3^{(4)}(t) \\ \ddot{F}_4(t) \end{pmatrix} = \begin{pmatrix} 1/m & 0 & 0 & 0 \\ 0 & g/J_1 & 0 & 0 \\ 0 & 0 & -g/J_2 & 0 \\ 0 & 0 & 0 & 1/J_3 \end{pmatrix} \begin{pmatrix} u_z \\ u_\theta \\ u_\phi \\ u_\psi \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} g$$

It can be seen that the vector of relative degrees is $r = (2, 4, 4, 2)$. The parameterization of the control inputs as function of flat output is then

$$\begin{pmatrix} u_z \\ u_\theta \\ u_\phi \\ u_\psi \end{pmatrix} = \begin{pmatrix} 1/m & 0 & 0 & 0 \\ 0 & g/J_1 & 0 & 0 \\ 0 & 0 & -g/J_2 & 0 \\ 0 & 0 & 0 & 1/J_3 \end{pmatrix}^{-1} \left[ \begin{pmatrix} \ddot{F}_1(t) \\ F_2^{(4)}(t) \\ F_3^{(4)}(t) \\ \ddot{F}_4(t) \end{pmatrix} - \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} g \right]$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

or

$$u_z(t) = m\left(\ddot{\mathcal{F}}_1(t) + g\right) \;;\; u_\theta(t) = J_1\frac{\mathcal{F}_2^{(4)}(t)}{g} \;;\; u_\phi(t) = -J_2\frac{\mathcal{F}_3^{(4)}(t)}{g} \;;\; u_\psi(t) = J_3\ddot{\mathcal{F}}_4(t) \quad (42)$$

Let $\mathcal{F}_i^*(t)$ be the reference trajectories for the flat outputs $\mathcal{F}_i(t)$ with $i = 1,\ldots,4$. As a consequence of the differential parameterization (41) and (42), if the system is forced to track the reference trajectories $\mathcal{F}_i^*(t)$, then the system states will be the following:

$$\begin{aligned} &x^*(t) = \mathcal{F}_2^*(t) \;;\; y^*(t) = \mathcal{F}_3^*(t) \;;\; z^*(t) = \mathcal{F}_1^*(t) \;; \\ &\theta^*(t) = \frac{\ddot{\mathcal{F}}_2^*(t)}{g} \;;\; \phi^*(t) = -\frac{\ddot{\mathcal{F}}_3^*(t)}{g} \;;\; \psi^*(t) = \mathcal{F}_4^*(t) \end{aligned} \quad (43)$$

and the nominal expected control inputs to be applied along the trajectories are as follows:

$$u_z^*(t) = m\left(\ddot{\mathcal{F}}_1^*(t) + g\right) \;;\; u_\theta^*(t) = J_1\frac{\mathcal{F}_2^{*(4)}(t)}{g} \;;\; u_\phi^*(t) = -J_2\frac{\mathcal{F}_3^{*(4)}(t)}{g} \;;\; u_\psi^*(t) = J_3\ddot{\mathcal{F}}_4^*(t) \quad (44)$$

In the rest of the paper, it is assumed that the quadrotor motion is restricted to the 2D plan (i.e., it is not changing altitude) and that it is not yawing. That is, $\mathcal{F}_1^*(t) = \text{constant}$ and $\mathcal{F}_4^*(t) = 0\ \forall t$. It turns out that the expected states and control inputs are as follows:

$$\begin{aligned} &x^*(t) = \mathcal{F}_2^*(t) \;;\; y^*(t) = \mathcal{F}_3^*(t) \;;\; z^*(t) = \text{constant}; \\ &\theta^*(t) = \frac{\ddot{\mathcal{F}}_2^*(t)}{g} \;;\; \phi^*(t) = -\frac{\ddot{\mathcal{F}}_3^*(t)}{g} \;;\; \psi^*(t) = 0 \end{aligned} \quad (45)$$

and

$$u_z^*(t) = mg \;;\; u_\theta^*(t) = J_1\frac{\mathcal{F}_2^{*(4)}(t)}{g} \;;\; u_\phi^*(t) = -J_2\frac{\mathcal{F}_3^{*(4)}(t)}{g} \;;\; u_\psi^*(t) = 0 \quad (46)$$

### *4.3. Trajectory parameterization*

Several methods can be used to design the reference trajectories $\mathcal{F}_i^*(t)$. The Bézier polynomial function [35] is employed in this paper. A general Bézier polynomial function of degree $n$ is the following:

$$\mathcal{F}_g(t) = a_n t^n + a_{n-1}t^{n-1} + \ldots + a_2 t^2 + a_1 t + a_0 \quad (47)$$

where $t$ is the time and $a_i$ $(i = 0,\ldots,n)$ are constant coefficients to be calculated in function of the initial and final conditions. It is clear that the larger is $n$, the smoother is the reference trajectory. However, calculations for trajectory planning become heavier as $n$ increases. For the quadrotor UAV, it can be seen in (44) that $u_\theta$ is in function of $\mathcal{F}_2^{(4)}(t)$ and $u_\phi$ is in function of $\mathcal{F}_3^{(4)}(t)$. The relative degrees are then $r_2 = 4$ and $r_3 = 4$. Smooth control inputs $u_\theta$ and $u_\phi$ can be obtained if one can impose $\mathcal{F}_2^*(t)$ and $\mathcal{F}_3^*(t)$ up to $\mathcal{F}_2^{*(4)}(t)$ and $\mathcal{F}_3^{*(4)}(t)$ at the initial and final time (i.e., up to $\mathcal{F}_2^{*(4)}(t_0)$, $\mathcal{F}_3^{*(4)}(t_0)$ and $\mathcal{F}_2^{*(4)}(t_f)$, $\mathcal{F}_3^{*(4)}(t_f)$). To this end, a Bézier polynomial function of degree 9 is employed for $\mathcal{F}_2^*(t)$ and $\mathcal{F}_3^*(t)$. The reference trajectories are then

$$\mathcal{F}_i^*(t) = a_9^i t^9 + a_8^i t^8 + \ldots + a_2^i t^2 + a_1^i t + a_0^i \;;\; (i = 2,3) \quad (48)$$

The coefficients $a_j^i$ $(i = 2,3$ and $j = 0,\ldots,9)$ are calculated to verify the initial conditions $\mathcal{F}_i^*(t_0)$, $\dot{\mathcal{F}}_i^*(t_0)$, $\ddot{\mathcal{F}}_i^*(t_0)$, $\mathcal{F}_i^{(3)*}(t_0)$, and $\mathcal{F}_i^{(4)*}(t_0)$ and the final conditions $\mathcal{F}_i^*(t_f)$, $\dot{\mathcal{F}}_i^*(t_f)$, $\ddot{\mathcal{F}}_i^*(t_f)$, $\mathcal{F}_i^{(3)*}(t_f)$, and $\mathcal{F}_i^{(4)*}(t_f)$. $t_0$ and $t_f$ are respectively the initial and final instants of the trajectory.

### *4.4. Trajectory planning using extrema analysis*

The trajectory planning consists in driving the quadrotor UAV from an initial position at time instant $t_0$ to a final position at time instant $t_f$ without violating system constraints. The initial and final conditions of the UAV as well as the initial time $t_0$ are all known, and the only unknown is the final time

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

of the mission $t_f$. Thus, the trajectory planning tends to tune the profile of the trajectory (by tuning $t_f$) so as to drive the system as fast as possible without violating system constraints. It can be seen from (46) that the nominal control inputs $u_z^*(t)$ and $u_\psi^*(t)$ are constant and independent of the reference trajectories. The trajectory planning consists then in determining $t_f$ while only considering $u_\theta^*(t)$ and $u_\phi^*(t)$ in the optimization problem:

$$P \begin{cases} \text{Minimize} & t_f - t_0 \\ \text{Subject to} & \underline{u}_\theta \leqslant J_1 \frac{\mathcal{F}_2^{*(4)}(t)}{g} \leqslant \overline{u}_\theta \\ & \underline{u}_\phi \leqslant -J_2 \frac{\mathcal{F}_3^{*(4)}(t)}{g} \leqslant \overline{u}_\phi \end{cases} \tag{49}$$

where $\underline{u}_\theta$ and $\underline{u}_\phi$ (respectively $\overline{u}_\theta$ and $\overline{u}_\phi$) are the lower bounds (respectively the upper bounds) for the control inputs $u_\theta(t)$ and $u_\phi(t)$. However, an optimization-based trajectory planning is computationally expensive and may not be adequate for real-time applications with limited calculation capabilities of on-board micro-computer. As an alternative, it is proposed in this paper to derive simple equations that can be used to instantaneously solve the trajectory planning problem. This consists in deriving the extrema of $u_i^*(t)$ $(i = 1, \ldots, 4)$ and then calculating $t_f$ so that the nominal desired control inputs are within the specified bounds. The detailed explanation can be found in Appendix A. As an example, the nominal value of the first PWM input $u_1^*(t)$ remains smaller than the upper bound $\overline{u}$ if $t_f$ is chosen as follows:

$$t_f \geqslant \left[ \left( \pm c_6 \frac{\mathcal{F}_2^*(t_f) J_1}{KLg\left(\overline{u} - \frac{mg}{4K}\right)} \right)^{\frac{1}{4}} ; \left( \pm c_7 \frac{\mathcal{F}_2^*(t_f) J_1}{KLg\left(\overline{u} - \frac{mg}{4K}\right)} \right)^{\frac{1}{4}} \right] \tag{50}$$

The solution obtained from the aforementioned equation involves aggressive maneuvers with big roll and pitch angles ($\phi$ and $\theta$). A controller based on the nonlinear model (36) would be able to guarantee the system's stability. However, a controller based on a linearized model (such as (38)) may not be able to keep system's stability when the quadrotor goes outside the linear operating zone. To avoid this latter situation, bounds are imposed on the maximal pitch and roll angles that are attained during the system's travel from the initial to the final position. The maximal pitch and roll angles are denoted as $\theta_{max}$ and $\phi_{max}$, respectively. As detailed in Appendix B, to guarantee that $|\theta^*(t)| \leqslant \theta_{max}$ and $|\phi^*(t)| \leqslant \phi_{max}$, it is sufficient to impose $t_f$ such as

$$t_f \geqslant \sqrt{\left| \frac{1215\sqrt{7}\mathcal{F}_2^*(t_f)}{343 g \theta_{max}} \right|} \quad \text{and} \quad t_f \geqslant \sqrt{\left| \frac{1215\sqrt{7}\mathcal{F}_3^*(t_f)}{343 g \phi_{max}} \right|} \tag{51}$$

The developed LQR with the integral action along with the trajectory planning have been applied to the quadrotor UAV testbed in the fault-free case. The trajectory planning is illustrated in Figure 4 with $\mathcal{F}_2(t_f) = -1.5$ m, $\mathcal{F}_3(t_f) = 0$, and initial conditions equal to zero. The figure shows the dynamic evolution of the height, which is set to 0.5 m, the pitch angle, the position of the system along the $x$-direction (where $\mathcal{F}_2 = x$ as pointed out in Section 4), and the PWM inputs to the four rotors. This experiment considers the case when $|\theta_{max}|$ is restricted to 15°. The trajectory tracking starts at time instant $t = 30$ s, and the total time of the mission is $t_f = 2.61$ s delimited by two vertical lines as shown in Figure 4. It is clear that when the system is forced to track the reference trajectory $\mathcal{F}_2^*(t)$, the pitch angle $\theta(t)$ remains within the interval $-15° \leqslant \theta(t) \leqslant +20°$ where the out of interval is due to model mismatch between the physical plant and the mathematical model (38) employed for trajectory planning. It can be concluded that thanks to flatness and the aforementioned developed approach, the system is driven as fast as possible to the desired setpoint of $\mathcal{F}_2(t_f) = -1.5$ m while respecting system constraints of $|\theta_{max}| = 15°$. The height regulation is slightly affected during the mission because the LQR controller provides good performance when the system is restricted to operate in the linear operating zone.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

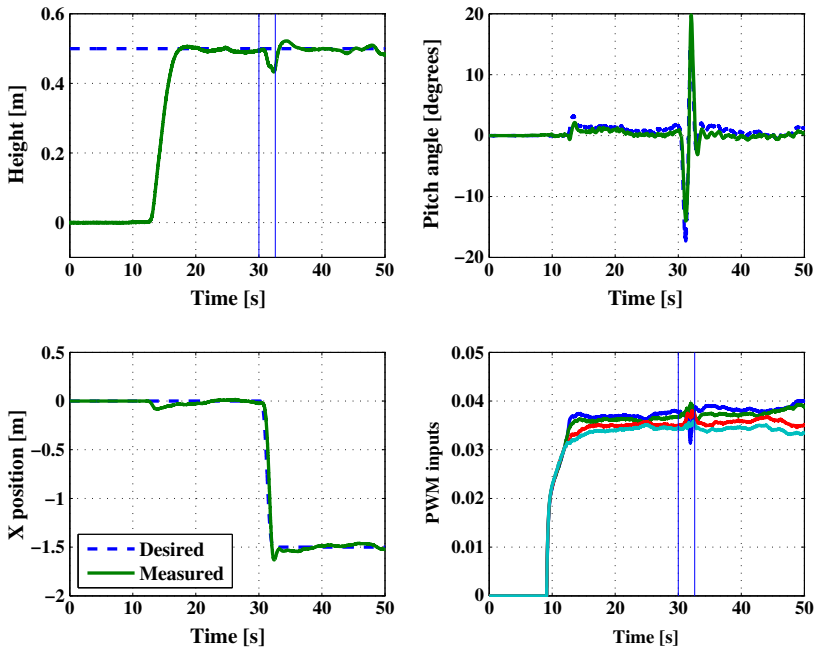


Figure 4. Trajectory planning in the nominal case with $|\theta_{max}| = 15°$. PWM, pulse-width modulation.

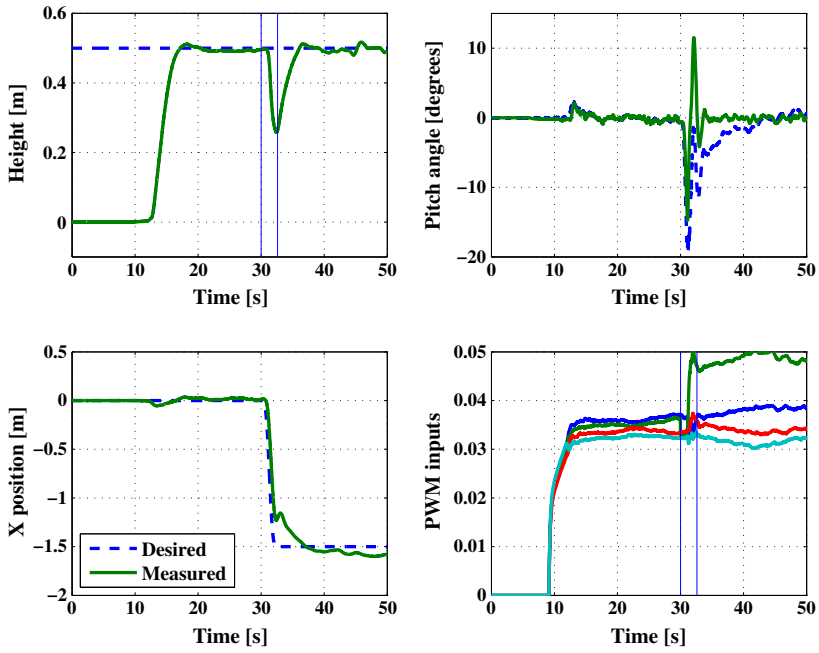


Figure 5. Trajectory planning in the case of 30% partial loss of control effectiveness in the second actuator ($w_2 = 0.3$) without fault-tolerant control. PWM, pulse-width modulation.

Figure 5 illustrates the loss of control effectiveness of 30% in the second motor (i.e., $w_2 = 0.3$) at time instant $t_{fault} = 31$ s (i.e., after 1 s of the mission's start). The actuator fault acts as a perturbation on the UAV system that loses its height and falls behind the reference trajectory. This sudden jump in the tracking error along $x$ and $z$ directions indicates an eventual occurrence of a fault. Because of the presence of the integral term in the controller, the system recovers and reaches again the desired trajectories. The controller reacts to the fault by increasing the second PWM input that is now close to the saturation limit of 0.05, and saturation occurs for a short time between 40

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

and 45 s. Because of the limited indoor laboratory space and the limited effective area covered by the OptiTrack cameras system, the quadrotor could not be required to travel for more than 1.5 m leading to a very short time of saturation. For outdoor applications when the system is required to travel for long distances, the saturation will occur for a longer time if the system is forced to track the pre-fault trajectory resulting in performance degradation and system's instability.

### *4.5. Active fault-tolerant control scheme without trajectory re-planning*

As shown in Figure 1, FDD is one of the primary functions in AFTCS [3]. Its goal is to perform two main decision tasks: fault detection, consisting of deciding whether or not a fault has occurred, and fault diagnosis, consisting of deciding which element of the system has failed and identifying the fault magnitude. For the Qball-X4 UAV, all state variables are available from the sensors with high measurement accuracy. Therefore, a state estimator for providing unknown states is not necessary in this case. However, in order to perform an efficient FDD for the partial loss of actuators control effectiveness represented by $w_i$ $(i = 1, \ldots, 4)$, together with the nonlinear state-space representation (36), a fault parameters identification scheme based on a UKF is employed for a parameter-estimation-based FDD, which is an extension of an earlier work developed for a fixed-wing NASA GTM UAV in [36]. In the fault-free case, the estimates of $w_i$ $(i = 1, \ldots, 4)$ are close to 0, and a deviation from 0 will be an indication for a fault occurrence. The UKF is a nonlinear extension of the Kalman filter, and it was developed to overcome some of the shortcomings of the extended Kalman filter in the estimation of nonlinear systems [37]. It should be noted that the strategy used in this paper is different from the traditional scheme that uses UKF for state or simultaneous state and parameter estimation [36], and it has the advantage of simplifying the FDD scheme to be more suitable for real-time applications. The parameter estimation problem, also referred to as system identification, involves determining a nonlinear mapping $y(t) = G(x(t), w(t))$, where $x(t)$ is the input, $y(t)$ is the output, and the nonlinear map $G$ is parameterized by the parameter vector $w$. Typically, a training set is provided with sample pairs consisting of known inputs and desired outputs, $\{x(t), d(t)\}$. The error is defined as $r(t) = d(t) - G(x(t), w(t))$, and the goal of learning involves solving for the parameters $w$ in order to minimize some given function of the error [38]. Under the assumption that step-wise actuator faults are assumed to occur, parameters can be efficiently estimated online by writing a new state-space representation as follows:

$$\begin{aligned} \dot{w}(t) &= \vec{0} + \nu(t) \\ d(t) &= G\left(x(t), w(t)\right) + r(t) \end{aligned} \tag{52}$$

where the parameters $w(t)$ correspond to a stationary process driven by process noise $\nu(t)$. The output $d(t)$ corresponds to a nonlinear observation on $w(t)$. The application of UKF to the Qball-X4 through system dynamics given by (36) provides the actuator fault magnitudes estimation $\hat{w}_i$ $(i = 1, \ldots, 4)$. To perform fault detection and decision making, many evaluation methods can be used to evaluate the difference between the estimated parameters $\hat{w}_i$ and the fault-free case when $w_i = 0$ such as the two-stage statistical hypothesis tests [25]. On the basis of the experimental data set given as an example, the actuator fault magnitudes corresponding to each of the four motors and their estimations are illustrated in Figure 6. As can be seen on such real experimental data, the fault magnitudes estimation is close to zero in the fault-free case and successfully identified the fault injected to the Qball-X4 with $w_2 = 0.3$ and $w_1 = w_3 = w_4 = 0$. The FDD module indicates which actuator is faulty and estimates the fault magnitude successfully.

On the basis of the information provided by the FDD module, the additive control law, as presented in (33), and the overall compensation control law (31) are computed in order to compensate the actuator fault effect on the system. With the fault accommodation method (Figure 7), the fault effect is reduced significantly with better tracking performance compared with the results shown in Figure 5 for the case without controller reconfiguration. With the actuator fault occurring at time $t_{fault} = 31$ s, the fault is diagnosed at $t_d = 31.5$ s (i.e., 100 sample times after fault occurrence), and the FTC is applied at $t_r = 32$ s (100 sample times after fault diagnosis). The system reaches the nominal values quicker because the fault is estimated and the new control law is able to compensate for the fault effect. The active FTC helps in recovering quicker the pre-fault performance before

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

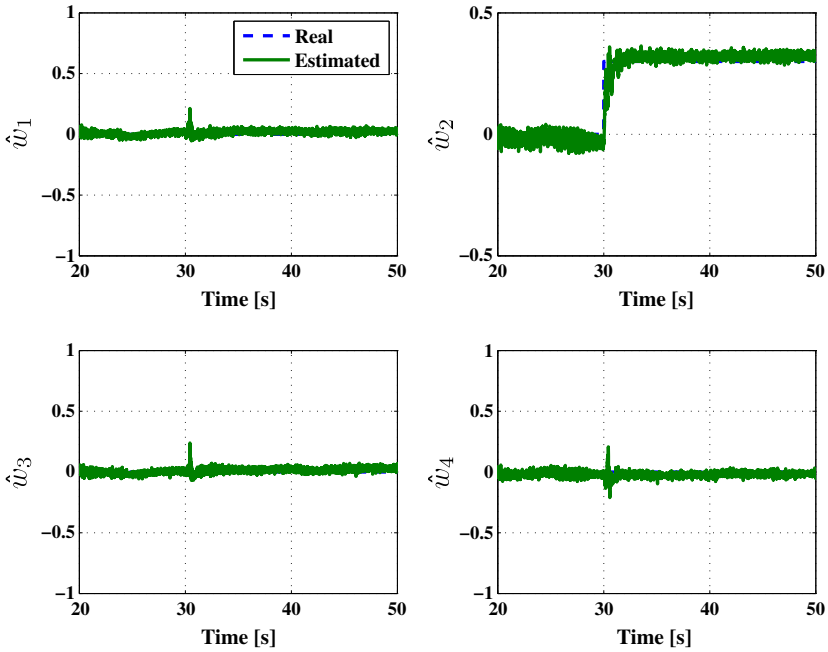


Figure 6. Fault identification with 30% partial loss of control effectiveness in the second actuator ($w_2 = 0.3$).

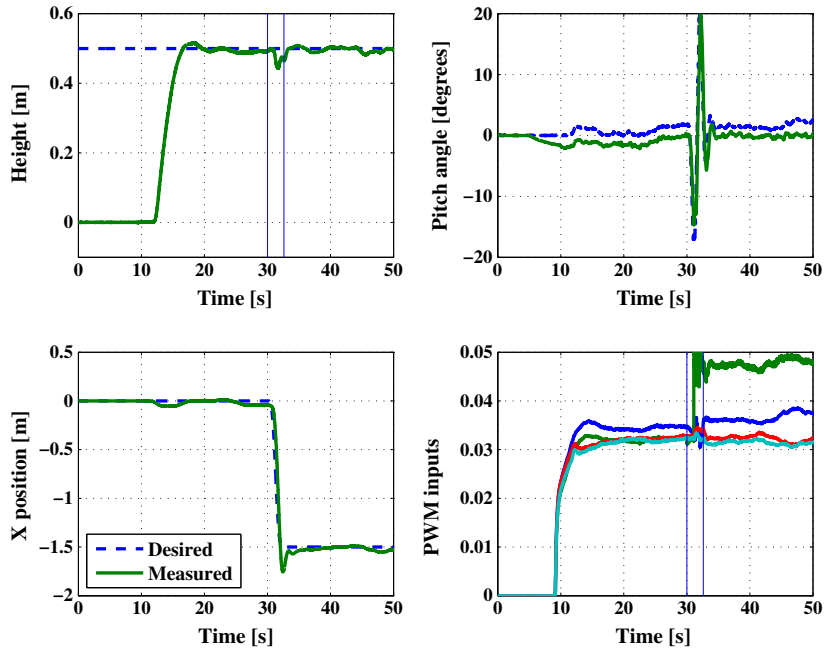


Figure 7. Active fault-tolerant control without trajectory re-planning under the fault case with $w_2 = 0.3$. PWM, pulse-width modulation.

the system goes unstable. However, as demonstrated in Figure 7, the total generated control input (for the second actuator) is similar to that without FTC (Figure 5). Thus, the FTC module does not provide a solution for the saturation problem. Therefore, to avoid saturation and preserve system's stability with the second control PWM input close to 0.05, a trajectory re-planning is required to re-plan the pre-fault trajectories based on the post-fault system conditions.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

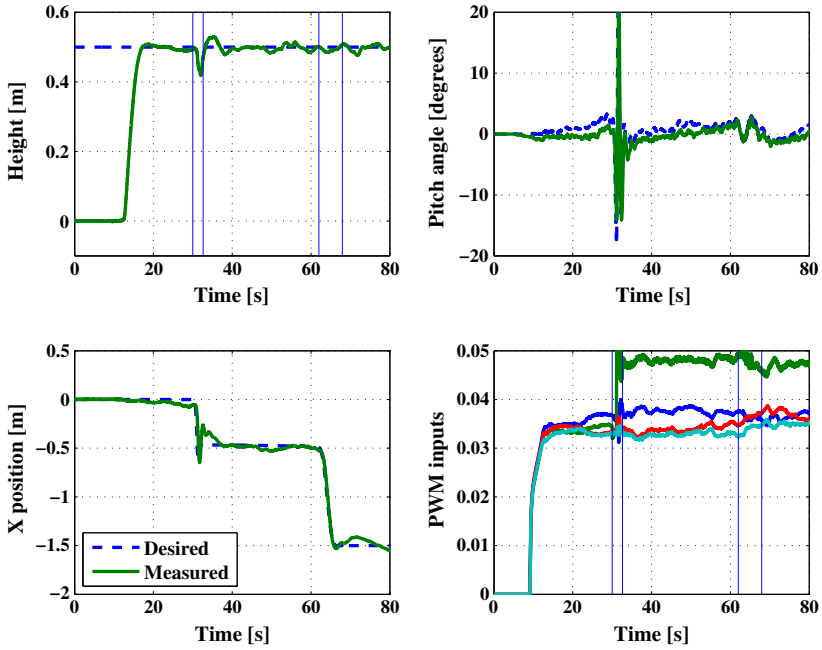


Figure 8. Active fault-tolerant control with trajectory re-planning under the fault case with $w_2 = 0.3$. PWM, pulse-width modulation.

### *4.6. Active fault-tolerant control with trajectory re-planning*

The testing result for the active FTC with trajectory re-planning is illustrated in Figure 8. Once the fault is detected and diagnosed at $t_d = 31.5$ s, the active FTC is applied to the system, and the trajectory is temporarily re-planned at time instant $t_{r1} = 32$ s ($t_{r1}$ is the time instant for the temporary trajectory). This temporary trajectory is performed using a second-order system where the natural frequency is set to a very small value and the damping ratio is set to one. This choice of natural frequency and damping ratio will result in a very slow reference trajectory serving in two directions. First, it will give sufficient time to have a better idea about the post-fault system conditions before taking any further action because the system will almost stay in hovering position. Second, the Qball-X4 moving velocity will drop to zero, and this will allow to easily re-plan the trajectory by following similar steps to those in Section 4.4. The temporary trajectory lasts (for illustration purpose) for about 30 s between $t_{r1} = 32$ s and $t_{r2} = 62$ s where the Qball-X4 is held in place before re-planning the trajectory at time instant $t_{r2} = 62$ s with $t_{f_{rep}} - t_{r2} = 5.94$ s ($t_{r2}$ is the time instant for the re-planned trajectory, and $t_{f_{rep}}$ is the final time of the mission with the re-planned trajectory). As can be seen in the figure, the re-planned trajectory is slower than the initial pre-fault trajectory to take into consideration the new actuator limits after fault occurrence. The PWM inputs are slightly excited during the re-planned trajectory, and as required, the second PWM input remains close to the upper limit of 0.05 but does not saturate. It can be concluded from this experiment that the flatness-based trajectory re-planning approach allows to take the system to its final desired destination as fast as possible in view of the post-fault system conditions. As before, the trajectory planning and re-planning intervals are delimited by two vertical lines as shown in Figure 8.

## 5. CONCLUSIONS

This paper proposes an AFTCS with a contribution of integrating an UKF-based FDD scheme, reconfigurable controller with an additive compensation control law, and a differential flatness-based trajectory re-planning strategy for handling actuator faults and saturation for improved tracking performance in the presence of actuator faults. The trajectory planning/re-planning is achieved

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

by solving simple equations where the objective is to determine adequate reference trajectories to reach the desired setpoints while respecting actuator saturation constraints. The proposed approach requires very little computational effort and is thus suitable for implementation to real-time systems with limited on-board computing capabilities. The approach has been experimentally tested and successfully applied to a quadrotor UAV testbed. Because of the limited space of the indoor laboratory and the limited area that can be covered by the OptiTrack cameras system, the quadrotor UAV was not able to travel for long distances, which has limited a full demonstration of the advantages of the proposed approach. Extension of the reconfiguration scheme developed in this work to other systems represents a future work of this research.

## APPENDIX A: EXTREMA CALCULATION FOR THE PULSE-WIDTH MODULATION INPUTS

This section explains how to determine the extrema for the PWM inputs. The system is required to move from an initial position $\mathcal{F}_i^*(t_0) = 0$ with initial conditions $\dot{\mathcal{F}}_i^*(t_0) = \ddot{\mathcal{F}}_i^*(t_0) = \mathcal{F}_i^{(3)*}(t_0) = \mathcal{F}_i^{(4)*}(t_0) = 0$ to a final position $\mathcal{F}_i^*(t_f) \neq 0$ with final conditions $\dot{\mathcal{F}}_i^*(t_f) = \ddot{\mathcal{F}}_i^*(t_f) = \mathcal{F}_i^{(3)*}(t_f) = \mathcal{F}_i^{(4)*}(t_f) = 0$ $(i = 2, 3)$. For $t_0 = 0$, (48) can be computed as follows:

$$\begin{aligned}\mathcal{F}_i^*(t) = 70\mathcal{F}_i^*(t_f)\frac{t^9}{t_f^9} - 315\mathcal{F}_i^*(t_f)\frac{t^8}{t_f^8} + 540\mathcal{F}_i^*(t_f)\frac{t^7}{t_f^7} \\ - 420\mathcal{F}_i^*(t_f)\frac{t^6}{t_f^6} + 126\mathcal{F}_i^*(t_f)\frac{t^5}{t_f^5} \;\; ; \;\; (i = 2, 3)\end{aligned} \tag{A.1}$$

Thus, it is possible to calculate $u_\theta^*(t) = J_1\mathcal{F}_2^{(4)*}(t)/g$ and $u_\phi^*(t) = -J_2\mathcal{F}_3^{(4)*}(t)/g$ because

$$\begin{aligned}\mathcal{F}_i^{(4)*}(t) = c_1\mathcal{F}_i^*(t_f)\frac{t^5}{t_f^9} - c_2\mathcal{F}_i^*(t_f)\frac{t^4}{t_f^8} + c_3\mathcal{F}_i^*(t_f)\frac{t^3}{t_f^7} \\ - c_4\mathcal{F}_i^*(t_f)\frac{t^2}{t_f^6} + c_5\mathcal{F}_i^*(t_f)\frac{t}{t_f^5} \;\; ; \;\; (i = 2, 3)\end{aligned} \tag{A.2}$$

where the $c_j$ $(j = 1, \ldots, 5)$ are constants that can easily be derived and are not given here for simplicity. From (37), (46), and (A.2), it is possible to determine the nominal PWM inputs to be applied along the reference trajectories. These nominal inputs are given as follows:

$$\begin{aligned}u_{1/2}^* &= \frac{mg}{4K} \pm \frac{\mathcal{F}_2^*(t_f)J_1\left(105840t^5 - 264600t_f t^4 + 226800t_f^2t^3 - 75600t_f^3t^2 + 7560t_f^4t\right)}{KLgt_f^9} \\ u_{4/3}^* &= \frac{mg}{4K} \pm \frac{\mathcal{F}_3^*(t_f)J_2\left(105840t^5 - 264600t_f t^4 + 226800t_f^2t^3 - 75600t_f^3t^2 + 7560t_f^4t\right)}{KLgt_f^9}\end{aligned} \tag{A.3}$$

It can be seen that when $t_f \to 0$, $u_i^* \to \infty$ and that when $t_f \to \infty$, $u_i^* \to mg/(4K)$. Thus, because $\mathcal{F}_2^*(t_f)$ and $\mathcal{F}_3^*(t_f)$ are known, one must find the minimal $t_f$ so that $\underline{u} \leq u^*(t) \leq \overline{u}$. One way to determine $t_f$ is to analytically calculate when the extrema of $u_i^*(t)$ $(i = 1, \ldots, 4)$ take place and what are their values (where the extrema collectively denote the maxima and the minima of a function). For this purpose, it is necessary to calculate the derivative of $u_i^*(t)$ in (A.3) with respect to time and then solve for $t$ the following equation:

$$\frac{du_i^*(t)}{dt} = 0 \;\; ; \;\; (i = 1, \ldots, 4) \tag{A.4}$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

For a fixed $i$ $(i = 1, \ldots, 4)$, each equation $du_i^*(t)/dt$ is a polynomial function of fourth degree and thus has four extrema that take place at

$$\frac{t_f}{2}\left(1 \pm \sqrt{\frac{3}{7} - \frac{2}{35}\sqrt{30}}\right) \tag{A.5}$$

and

$$\frac{t_f}{2}\left(1 \pm \sqrt{\frac{3}{7} + \frac{2}{35}\sqrt{30}}\right) \tag{A.6}$$

The values of the four extrema can be obtained by replacing $t$ of (A.3) by the values obtained in (A.5) and (A.6). As an example, the values of the four extrema of $u_1^*(t)$ of the first rotor are simply

$$u_{1_{Ext}}^* = \left[\frac{mg}{4K} \pm c_6 \frac{\mathcal{F}_2^*(t_f) J_1}{KLgt_f^4} \;\; ; \;\; \frac{mg}{4K} \pm c_7 \frac{\mathcal{F}_2^*(t_f) J_1}{KLgt_f^4}\right] \tag{A.7}$$

with $c_6$ and $c_7$ two constants. The determination of $t_f$ consists finally in solving (A.7) so that the extrema of the nominal PWM input verify

$$\underline{u} \leqslant u_{i_{Ext}}^* \leqslant \overline{u} \;\; ; \;\; i = 1, \ldots, 4 \tag{A.8}$$

By considering (A.7) and the upper bound $\overline{u}$ in (A.8), four solutions are obtained:

$$t_f \geqslant \left[\left(\pm c_6 \frac{\mathcal{F}_2^*(t_f) J_1}{KLg\left(\overline{u} - \frac{mg}{4K}\right)}\right)^{\frac{1}{4}} ; \left(\pm c_7 \frac{\mathcal{F}_2^*(t_f) J_1}{KLg\left(\overline{u} - \frac{mg}{4K}\right)}\right)^{\frac{1}{4}}\right] \tag{A.9}$$

The four extrema of $u_2^*(t)$, $u_3^*(t)$, and $u_4^*(t)$ have the same structure as for $u_1^*(t)$ with different signs and different values for the constants $c_6$ and $c_7$. In conclusion, the quadrotor UAV will have 16 extrema (four per motor). The trajectory planning consists then in solving $t_f$ so that (A.8) is verified. Thus, 32 solutions are obtained (for the lower and upper bounds), and the one to consider is the maximal among these solutions.

## APPENDIX B: EXTREMA CALCULATION FOR THE ANGLES

According to (41), the nominal pitch and roll angles that the system will experience when it is forced to track the reference trajectories are as follows:

$$\theta^*(t) = \frac{\ddot{\mathcal{F}}_2^*(t)}{g}; \; \phi^*(t) = -\frac{\ddot{\mathcal{F}}_3^*(t)}{g}. \tag{B.1}$$

Similarly as before, it is possible to determine when the extrema $\theta_{Ext}^*$ and $\phi_{Ext}^*$ are occurring and what are their values. The minima are zero, whereas the maxima are taking place at

$$\left(\frac{1}{2} + \frac{\sqrt{7}}{14}\right) t_f \;\; \text{and} \;\; \left(\frac{1}{2} - \frac{\sqrt{7}}{14}\right) t_f \tag{B.2}$$

and their values are

$$\theta_{Ext}^* = \pm \frac{1215\sqrt{7}}{343} \frac{\mathcal{F}_2^*(t_f)}{gt_f^2} \;\; \text{and} \;\; \phi_{Ext}^* = \pm \frac{1215\sqrt{7}}{343} \frac{\mathcal{F}_3^*(t_f)}{gt_f^2} \tag{B.3}$$

To guarantee that

$$\left|\theta_{Ext}^*\right| \leqslant \theta_{max} \;\; \text{and} \;\; \left|\phi_{Ext}^*\right| \leqslant \phi_{max} \tag{B.4}$$

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

where $\theta_{max}$ and $\phi_{max}$ are the maximal allowed pitch and roll angles, it is sufficient to impose $t_f$ such as

$$t_f \geqslant \sqrt{\left|\frac{1215\sqrt{7}\mathcal{F}_2^*(t_f)}{343g\theta_{max}}\right|} \text{ and } t_f \geqslant \sqrt{\left|\frac{1215\sqrt{7}\mathcal{F}_3^*(t_f)}{343g\phi_{max}}\right|} \tag{B.5}$$

ACKNOWLEDGEMENTS

This work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through a Postdoctoral Fellowship (PDF) program, an NSERC Strategic Project Grant, and an NSERC Discovery Project Grant. Support from Quanser Inc. and colleagues from Quanser Inc. for the development of the Qball-X4 UAV testbed is also highly appreciated.

REFERENCES

1. Harris TJ, Seppala C, Desborough LD. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *Automatica* 1999; **38**(12):2063–2073.
2. Patton RJ. Fault-tolerant control systems: the 1997 situation. *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, Vol. 2, Kingston Upon Hull, United Kingdom, 1997; 1033–1055.
3. Zhang YM, Jiang J. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control* 2008; **32**(2):229–252.
4. Mahmoud M, Jiang J, Zhang YM. *Active Fault Tolerant Control Systems: Stochastic Analysis and Synthesis*. Springer-Verlag: Berlin, Heidelberg, 2003.
5. Blanke M, Kinnaert M, Lunze J, Staroswiecki M. *Diagnosis and Fault-tolerant Control*. Springer-Verlag: Berlin, Heidelberg, 2006.
6. Isermann R. *Fault-diagnosis Systems: An Introduction From Fault Detection to Fault Tolerance*. Springer-Verlag: Berlin, Heidelberg, 2006.
7. Noura H, Theilliol D, Ponsart JC, Chamseddine A. *Fault-tolerant Control Systems: Design and Practical Applications*. Springer-Verlag: London, 2009.
8. Ducard G. *Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*. Springer-Verlag: London, 2009.
9. Edwards C, Lombaerts T, Smaili H. *Fault Tolerant Flight Control: A Benchmark Challenge*. Springer-Verlag: Berlin, Heidelberg, 2010.
10. Alwi H, Edwards C, Tan CP. *Fault Detection and Fault-tolerant Control Using Sliding Modes*. Springer-Verlag: London, 2011.
11. Jiang J, Yu X. Fault-tolerant control systems: a comparative study between active and passive approaches. *Annual Reviews in Control* 2012; **36**(1):60–72.
12. Li T, Zhang YM, Gordon BW. Passive and active nonlinear fault-tolerant control of a quadrotor UAV based on sliding mode control technique. *Proceedings of the Institution of Mechanical Engineers, Part I - Journal of Systems and Control Engineering* 2013; **227**(1):12–23.
13. Bemporad A, Casavola A, Mosca E. Nonlinear control of constrained linear systems via predictive reference management. *IEEE Transactions on Automatic Control* 1997; **42**(3):340–349.
14. Gilbert E, Kolmanovsky I. Nonlinear tracking control in the presence of state and control constraints: a generalized reference governor. *Automatica* 2002; **38**(12):2063–2073.
15. Zhang YM, Jiang J. Fault tolerant control system design with explicit consideration of performance degradation. *IEEE Transactions on Aerospace and Electronic Systems* 2003; **39**(3):838–848.
16. Zhang YM, Jiang J, Theilliol D. Incorporating performance degradation in fault tolerant control system design with multiple actuator failures. *International Journal of Control, Automation, and Systems* 2008; **6**(3):327–338.
17. Theilliol D, Join C, Zhang YM. Actuator fault tolerant control design based on reconfigurable reference input. *International Journal of Applied Mathematics and Computer Science* 2008; **18**(4):553–560.
18. Dardinier-Maron V, Hamelin F, Noura H. A fault-tolerant control design against major actuators failures: application to a three-tank system. *Proceedings of the 38th IEEE Conference on Decision and Control*, Phoenix, AZ, USA, 1999; 3569–3574.
19. Chamseddine A, Join C, Theilliol D. Trajectory planning/re-planning for satellite systems in rendezvous mission in the presence of actuator faults based on attainable efforts analysis. *International Journal of Systems Science* 2013. DOI: 10.1080/00207721.2013.797034.
20. Fliess M, Lévine J, Martin P, Rouchon P. Flatness and defect of nonlinear systems: introductory theory and example. *International Journal of Control* 1995; **61**(6):1327–1361.
21. Gensior A, Woywode O, Rudolph J, Güldner H. On differential flatness, trajectory planning, observers, and stabilization for DC-DC converters. *IEEE Transactions on Circuits and Systems-I* 2006; **53**(9):2000–2010.
22. Tsourdos A, White BA, Shanmugavel M. *Cooperative Path Planning of Unmanned Aerial Vehicles*, Progress in Astronautics and Aeronautics Series. John Wiley & Sons: Chichester, 2010.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

23. Mueller M. Normal form for linear systems with respect to its vector relative degree. *Linear Algebra and its Applications* 2009; **430**(4):1292–1312.
24. Wu NE, Zhang YM, Zhou KM. Detection, estimation, and accommodation of loss of control effectiveness. *International Journal of Adaptive Control and Signal Processing* 2000; **14**(7):775–795.
25. Zhang YM, Jiang J. An active fault-tolerant control system against partial actuator failures. *IEE Proceedings Control Theory and Applications* 2002; **149**(1):95–104.
26. Noura H, Sauter D, Hamelin F, Theilliol D. Fault-tolerant control in dynamic systems: application to a winding machine. *IEEE Control Systems Magazine* 2000; **20**(1):33–49.
27. Theilliol D, Noura H, Ponsart JC. Fault diagnosis and accommodation of a three-tank system based on analytical redundancy. *ISA Transactions* 2002; **41**(3):365–382.
28. Rodrigues M, Theilliol D, Aberkane S, Sauter D. Fault tolerant control design for polytopic LPV system. *International Journal of Applied Mathematics and Computer Science* 2007; **17**(1):27–37.
29. Boskovic JD, Prasanth R, Mehra RK. Retrofit fault-tolerant flight control design under control effector damage. *Journal of Guidance, Control, and Dynamics* 2007; **30**(3):703–712.
30. Guenab F, Weber P, Theilliol D, Zhang YM. Design of a fault tolerant control system incorporating reliability analysis under dynamic behaviour constraints. *International Journal of Systems Science* 2011; **42**(1):219–223.
31. Gao Z, Antsaklis PJ. Stability of the pseudo-inverse method for reconfigurable control systems. *International Journal of Control* 1991; **53**(3):717–729.
32. Staroswiecki M. Fault tolerant control: the pseudo-inverse method revisited. *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, 2005.
33. Xu R, Ozguner U. Sliding mode control of a quadrotor helicopter. *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006; 4957–4962.
34. D'Azzo J, Houpis CH. *Linear Control System Analysis and Design: Conventional and Modern*, McGraw-Hill Series in Electrical and Computer Engineering. McGraw-Hill Companies: New York, 1995.
35. Mai P, Join C, Reger J. Flatness-based fault-tolerant control of a nonlinear MIMO system using algebraic derivative estimation. *Preprints of the 3rd IFAC Symposium on System, Structure and Control*, Foz do Iguassu, Brazil, 2007; 350–355.
36. Ma L, Zhang YM. DUKF-based fault detection and diagnosis for GTM UAV using nonlinear and LPV models. *Proceedings of the 6th ASME/IEEE International Conference on Mechatronic and Embedded Systems and Applications*, Qingdao, P. R. China, 2010; 375-380.
37. VanDyke M, Schwartz J, Hall C. Unscented Kalman filtering for spacecraft attitude state and parameter estimation. *Proceedings of the 14th AAS/AIAA Space Flight Mechanics Meeting*, Maui, Hawaii, 2004.
38. van der Merwe R, Wan E. The square-root unscented Kalman filter for state and parameter estimation. *Proceedings of the International Conference on Acoustics, Speech and Signal Process (ICASSP'01)*, Vol. 6, Salt Lake City, Utah, USA, 2001; 3461–3464.

Copyright © 2013 John Wiley & Sons, Ltd.

DOI: 10.1002/acs

10991115, 2015, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/acs.2451 by Huazhong University Of Sci & Tech, Wiley Online Library on [09/04/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License