UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSGRADO

Fault Tolerant Control By Flatness Approach

por

César Martínez Torres

en opción al grado de

Doctor en Ingeniería eléctrica

con especialidad en control

San Nicolás de los Garza, Nuevo León Diciembre 2013

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSGRADO

Fault Tolerant Control By Flatness Approach

por

César Martínez Torres

en opción al grado de

Doctor en Ingeniería eléctrica

con especialidad en control

San Nicolás de los Garza, Nuevo León Diciembre 2013

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSGRADO

Los miembros del Comité de Tesis recomendamos que la Tesis «Fault Tolerant Control By Flatness Approach», realizada por el alumno César Martínez Torres, con número de matrícula 1209182, sea aceptada para su defensa como opción al grado de Doctor en Ingeniería eléctrica con especialidad en Control.

El Comité de Tesis

Dr. Franck Cazaurang Asesor	Dr. Efraín Alcorta García Asesor
Dr. Loïc Lavigne Co-asesor	Dr. David A. Díaz Romero Co-asesor
Dr. Hebertt Sira Ramírez Revisor	Dr. Didier Theilliol Revisor

San Nicolás de los Garza, Nuevo León, Diciembre 2013

ACKNOWLEDGEMENTS

I would like to express mi gratitude to mi supervisors Dr. Franck Cazaurang and Dr. Efraín Alcorta García, as well as to mi co-supervisors Dr. Loïc Lavigne and Dr. David A. Díaz Romero for the time that they consecrated to my PhD project.

In memoriam of my Granny

Paula Casillas López.

The greater our knowledge increases

the more our ignorance unfolds.

John F. Kennedy

CONTENTS

ABSTRACT

The objective of this PhD work is provide a flatness based active fault tolerant control technique. For such systems, it is possible to find a set of variables, named flat outputs such that states and control inputs can be expressed as functions of flat outputs and their time derivatives. The fault detection and isolation block has to provide a fast and accurate fault isolation, this action is carried out by exploiting the non-uniqueness property of the flat outputs, in fact if a second set of flat outputs algebraically independent and differentially coupled of the first are found, the number of residues augments. By consequence this could help to isolate more faults than if only one set is found.

Regarding reconfiguration if the flat system counts with the properties listed above we will obtain versions of states and control inputs as much of flat outputs vector are found, because each control input and state is function of the flat output. The proposed approach provides in this manner one measure related to a faulty flat output vector and one or more computed by using and unfaulty one.

The redundant state signals could be use as reference of the controller in order to hide the fault effect. This will be helpful to provide an entirely flatness-based fault tolerant control strategy.

The works presented in this manuscript are under the next hypothesis:

- The flat outputs are always states of the system or a linear combination of them.

- The control loop is closed with a state feedback controller.
- The states included in the flat output vector are measured or at least estimated.
- Faults affecting the actuators are considered rejected by the controller, by consequence reconfiguration is only carried out after sensor faults.

Since a flat system could be linear or nonlinear, the proposed approach could be applied in either of two type of systems. Feasibility of this approach is analyzed in two nonlinear plants, an unmanned quadrotor and a three tank system.

General introduction

In the last decades, demographic explosion and globalization, unchained the necessity to design and operate profitable productions process and reliable transport systems. The presence of a fault in a production process can lead to substantial loss, not only in the manufactured product, but in the production equipment itself. In some systems, the fault occurrence is even more critical, for example if an airplane flying at cruising attitude (35000 ft.) is affected by a fault, the consequences of it can lead the airplane to destruction and by consequence the lost of human lives. Nowadays airplanes count with a surveillance stage, which is in charge of monitor the entire system and assure the safety of the vehicle. In order to accomplish such task, manufacturers use physical redundancy, meaning that two or more subsystems (e.g. flight-control computers, sensors) work in the same time, this action permits to be robust again a simple fault and guarantees a failure rate lower than 10^{-9} failure per flight hour. This solution is easy to implement but represents a high cost due to the necessity to triple or quadruple the elements.

In complex systems (e.g. Nuclear plants, Petrochemical process, airplanes), every single component has been designed to accomplish a particular task, in order to permit the global operation of the system. Thus, a failure appearing in actuators, sensors or the system itself may affect the nominal performance. The classical control techniques assure the system stability in closed loop and the nominal performance desired if no-fault is present. However in a faulty case a classic closed loop may result in a low performance or system instability and by consequently the possible system destruction.

In order to avoid system lost, researchers developed control systems capable of self-repair, meaning that, the controller assure at least system stability and at the best nominal behavior despite the apparition of a fault. Systems presenting this capability are known as Fault Tolerant Control Systems (FTC). Nowadays fault reconfiguration make an essential part of almost every controlled system. FTC systems are designed to behave as a classical control system until a fault affects the controlled plant. If the plant is affected by a fault, the FTC system has to be capable of detect, identify and reject it as soon as possible. Such actions have as final objective preserve as the best nominal behavior and at least stability.

During all the long of this manuscript a fault is defined as in the book Fault-diagnois systems published by Rolf Isermann, [34]. Such publication defines the fault as “an unpermitted deviation of at least one characteristic property of the system from the acceptable, usual, standard condition”. Two different types of faults are considered, additive and multiplicative. The first one is represented by the addition of a term in the measure or in the control input according to the current case, see equation (1a). Multiplicative faults are represented by a term which multiplies the measure or the input according to the affected variable. See equation (1b).

$$Y(t) = U(t) + f(t) \tag{1a}$$

$$Y(t) = (A + f(t))(U(t)) \tag{1b}$$

Where $Y(t)$ represents the output of the sensor or the actuator, $U(t)$ stands for the sensor or actuator input signal. $f(t)$ represents the fault. A denotes a multiplicative factor which is usually equal to one. Fig. 1 shows the block diagrams corresponding to each kind of fault.

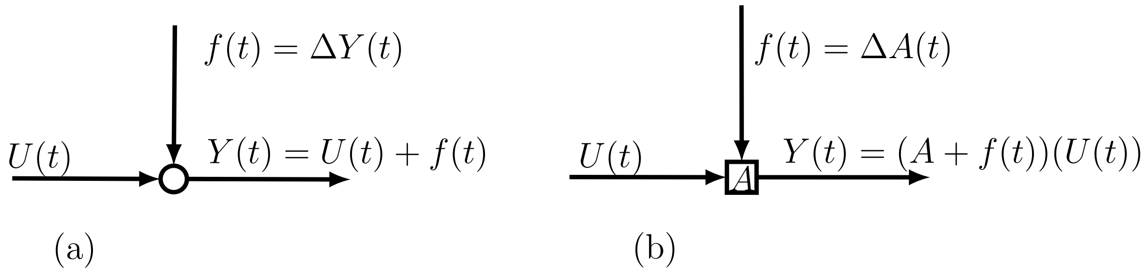

Figure 1: (a) Additive fault; (b) Multiplicative fault

In order to counteract the fault, two different strategies could be used, passive and active, the first strategy comes from the group of robust control. [3]. The active approaches are characterized by the presence of a Fault Detection and Isolation (FDI) module, which in function of the fault send information to the reconfiguration block in order to adapt the system to counteract the fault effect. See section 1.4.1 for more details.

The works presented in this manuscript fit into the framework of active methods. The proposed approach use the properties of the differentially flat systems to generate analytical redundancy, such redundancy can be use to generate residual signals. The main characteristic of the proposed approach is the fact that the FDI module is coupled with the reconfiguration block, this action could reduce the computational charge by minimizing the reaction time to counteract the fault. Besides, because the approach is based on the properties of the differentially flat systems it could be applied to linear and nonlinear systems indistinctly. This work is devoted to investigate feasibility on nonlinear systems.

The manuscript is divided in three chapters:

The chapter number 1 is devoted to present the properties and the definition of the flat systems, the flatness-based motion planning is presented as well. A state of the art of the technique of Fault detection isolation and reconfiguration is developed as well.

The chapter number 2 presents the proposed approach. In first time a state of the art in flatness-based FTC approaches presented in the literature is commented. The algorithm to compute the flat output is presented. In order to facilitate the comprehension and highlight the advantages of the technique the fault tolerant control technique is divided in the fault detection and isolation task and the reconfiguration block. Both activities are divided in two cases, in order to show that if a flat system has at least two different set of flat outputs the FTC method is improved.

The chapter number 3 is dedicated to show the applicability of the FTC method. Two systems are taken into account, the first one is a UAV quadrotor, this system only count with a set of flat outputs, however partial reconfiguration could be applied. In a second time the technique is applied in a classical three tank system, such plant in contrast to the UAV present two set of flat outputs, this feature is exploited to full-reconfigure the system after fault.

CHAPTER 1

FUNDAMENTALS

Abstract:

The goal of this chapter is to present the basic concepts of the main parts of this research work. The properties of the so-called differential flatness systems are presented. Flatness-based motion planning is presented as well. The definition of the Fault Tolerant Control systems is developed in section 1.4. Sections 1.5 and 1.6 are devoted to present the model-based Fault detection and Isolation techniques and the existent fault reconfiguration techniques.

1.1 INTRODUCTION

The appearance of a fault in a system affects directly its performance, by consequence this will impact the final objective of the system e.g., final position of a control surface in a plane, water level in a tank, etc. The classical control laws are designed to ensure stability and nominal performance of the system. However a classic controller don't take into account the apparition of faults affecting sensors, actuators or the system itself, such appearance will affects the nominal performance in the best of cases, and in the worst one the system will lost not only performance but even stability. Such behavior should be avoided, especially in critical systems, for instance nuclear plants or airplanes.

Control systems that take into account such scenario are known as Fault Tolerant Control Systems (FTC). Those systems can globally be divided in two main tasks: Fault Detection and Isolation (FDI) and Control Reconfiguration.

This chapter is devoted to present such control systems and some techniques of fault detection and fault recovery. Regarding FDI, special attention is dedicated to model-based approaches [18] whose fault detection principle is based in the comparison between sensor measures and the measure estimation coming from a mathematical model describing the physical process. For fault recovery, the research is focused in control reconfiguration [47].

The main contribution of this thesis is based on the properties of the so-called differentially flat systems [23]. The next section presents the definition of those particular systems, as well as the flatness-based motion planning approach. Which, is facilitated thanks to the inherent properties of the flat systems. Section 1.4 presents the different approaches presented in the literature for FTC systems, main attention is focused on active systems, which reacts after a fault occurrence, in order to prevent system loss.

Section 1.5 is devoted to FDI systems and particularly for quantitative model-based methods. A general outlook of these methods is presented as well. A non-exhaustive list of fault recovery methods is presented in section 1.6.

1.2 DIFFERENTIALLY FLAT SYSTEMS

The flatness theory search to determine if a system of differential equations could be parametrized by arbitrary functions. The first works have been carried out in [10], aiming aeronautical applications. The theory development continued in the Phd dissertation of P. Martin [50], this work has lead to the formal concept of flatness presented by M. Fliess et al. in [23].

The differential flatness of non-linear and linear systems, could be described by using mathematical formalisms, and specifically differential algebra or differential geometry.

1.2.1 FLATNESS CONCEPT

A non-linear or linear system is flat if there exists a set of variables differentially independents, called flat outputs, whose number is equal to the quantity of control inputs, such as, the vector state and the control inputs can be expressed as functions of the flat outputs and a finite number of its time derivatives. By consequence, state and control inputs trajectories can be obtained by planning only the flat output trajectories, this property can be particularly exploited on trajectory planning, see [44,45,57,73] and trajectory tracking [2,71]. Flatness could be used to design robust controllers, see for instance tesis de loic y franck

Definition 1.1 *Flat system:*

Let us consider the nonlinear system $\dot{x} = f(x,u)$, $x \in \Re^n$ the state vector, $u \in \Re^m$ the control vector and f a C^∞ function of x and u. The system is differentially flat if, and only if, it exists a flat output vector $z \in \Re^m$ such as:

- The flat output vector its expressed as function of the state x and the control input u and a finite number of its time derivatives.

$$z = \phi_z(x, u, \dot{u}, ..., u^{(\gamma)}) \tag{1.1}$$

- The state x and the control input u are expressed as functions of the vector z and a finite number of its time derivatives.

$$\begin{aligned} x &= \phi_x(z, \dot{z}, ..., z^{(\alpha)}) & (1.2) \\ u &= \phi_u(z, \dot{z}, ..., z^{(\alpha+1)}) & (1.3) \end{aligned}$$

Where $z^{(\alpha)}$ denotes the α^{th} time derivative of z.

Every flat system is equivalent to a linear controllable one by diffeomorphism and endogenous dynamic feedback, moreover, the flat outputs are the solutions of the system of differential equations which determines the diffeomorphism and the feedback linearizable, by consequence, every controllable linear system is flat, and conversely. Moreover regarding observability a flat system is always observable from the flat outputs.

1.2.2 Flat systems examples

This section presents various examples of flat systems. Additional examples can be found in [40].

Example 1.2 *Planar ducted fan [73]:*

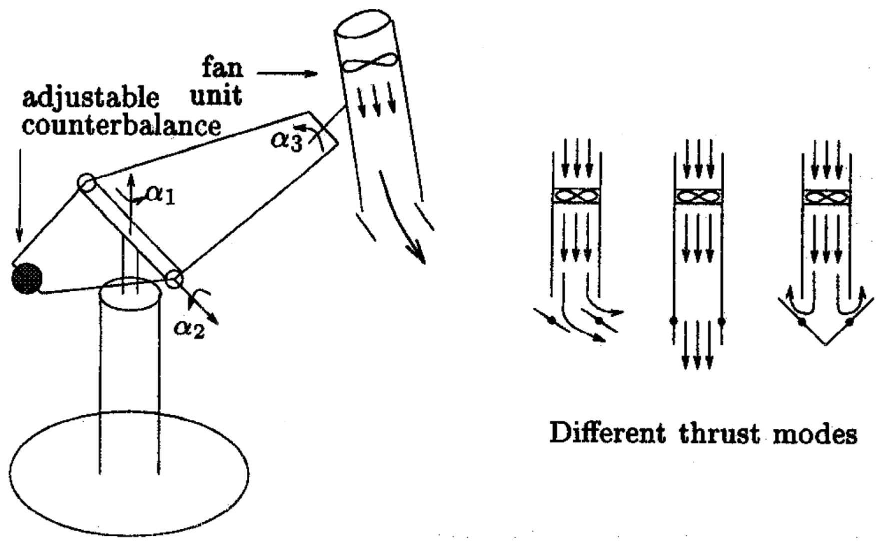

Figure 1.1: Planar ducted fan [73]

The system is mounted on a rotating arm that moves in as the fan moves up. [36], see Fig. 1.1 neglecting some dynamics the nonlinear model obtained is:

$$\begin{bmatrix} m_x\dot{x} \\ m_y\dot{y} \\ J\dot{\theta} \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \\ r & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 + m_g g \end{bmatrix} + \begin{bmatrix} 0 \\ -m_g g \\ 0 \end{bmatrix} \tag{1.4}$$

Where (x and y) are the coordinates of the center of mass, θ is the angle with the vertical axis, u_1 is the force perpendicular to the fan shroud, r is the distance between the center of mass and the point where the force is applied, g is the gravitational constant, m_x and m_y are the inertial mass of the fan in the (x, y) direction respectively, $m_g g$ is the weight of the fan, and J is the moment of inertia. The tracking outputs are the (x, y) coordinates of the center of mass.

The flat outputs are:

$$z_1 = x - \frac{J}{m_x r} sin\theta \tag{1.5}$$

$$z_2 = y - \frac{J}{m_y r} cos\theta \tag{1.6}$$

The angle θ can be expressed in function of the flat outputs:

$$\theta = \frac{-m_x\ddot{z}_1}{m_y\ddot{z}_2 + m_g g} \tag{1.7}$$

Using the equations above is straightforward to found the expressions of the states. Interested reader could found calculation details in [17] and [51]. After obtaining the necessary time derivatives, each control input could be expressed as function of the flat outputs.

$$u_1 = \frac{J\ddot{\theta}}{r} \tag{1.8}$$

$$u_2 = \frac{cos\theta u_1 - m_x\ddot{x} - sin\theta m_g g}{sin\theta} \tag{1.9}$$

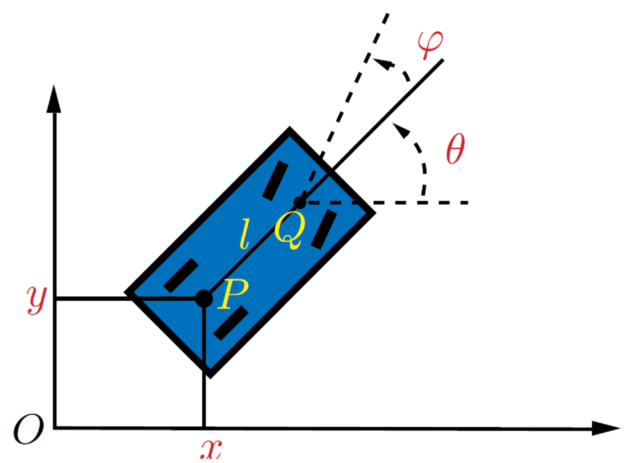

Figure 1.2: Non holonomic car [40]

Example 1.3 *Non holonomic car [40]:*

Consider a vehicle of four wheels rolling without slipping on the horizontal plane. We denote by (x, y) the coordinates of the point P, middle of the rear axle, Q the middle point of the front axis, θ the angle between the longitudinal axis of the vehicle and the Ox axis. The system counts with two control inputs u which denotes the vertical thrust and φ the angle of the front wheels. See Fig. 1.2.

The mathematical model of the vehicle can be expressed as follows:

$$\begin{aligned} \dot{x} &= ucos\theta \\ \dot{y} &= usin\theta \\ \dot{\theta} &= \frac{u}{l}tan\varphi \end{aligned} \tag{1.10}$$

This system has two control inputs, by definition the flat output vector is compound of two elements. Let us prove that $z = [x,\ y]^T$.

Combining the two first equations of (1.10) we can obtain:

$$\begin{aligned} \theta &= tan^{-1}\left(\frac{\dot{z}_2}{\dot{z}_1}\right) \\ u &= \sqrt{\dot{z_1}^2 + \dot{z_2}^2} \end{aligned} \tag{1.11}$$

The expression of $\dot{\theta}$ could be found by computing the time derivative of the first equation in (2.14). Such computation leads to:

$$\dot{\theta} = \frac{\ddot{z}_2\dot{z}_1 - \dot{z}_2\ddot{z}_1}{\dot{z_1}^2 + \dot{z_2}^2} \tag{1.12}$$

From the third equation of (1.10) we can compute

$$\varphi = tan^{-1}\left(\frac{l\dot{\theta}}{u}\right) = tan^{-1}\left(\frac{(l(\ddot{z}_2\dot{z}_1 - \dot{z}_2\ddot{z}_1)}{(\dot{z_1}^2 + \dot{z_2}^2)^{\frac{3}{2}}}\right) \tag{1.13}$$

Equations (2.14,2.15,1.13), demonstrate that the non holonomic car described by the equations (1.10) is flat.

1.3 MOTION PLANNING

The goal of motion planning is to find control actions that moves the concerned system from a start state to a goal condition, while respecting constraints and avoiding collision.

Motion planning for manipulators robots have attracted research interest in the beginnings of the 90's, see [46] for instance. More recently special attention is devoted to vehicle motion planning, which is a special case, such increases in solving difficulty when the degrees of freedom (DOF) augments. See for instance [29] and references therein for an extensive planning algorithms survey for UAV's.

This section is focused in the flatness-based motion planning approach.

1.3.1 TERMINOLOGY

The terminology [38] used in this work is the next:

- **Path planning:** A geometric representation to move from an initial to a final condition. The main goal is to find a collision-free path among a collection of static and dynamic obstacles.
- **Trajectory planning:** also known as trajectory generation. It includes velocities, accelerations, and jerks along the path. Normally the main task is to find trajectories for a priori specified paths. Those trajectories could be obliged to fulfill a certain criterion (eg., minimum execution time, minimum energy consumption).
- **Motion planning:** Is the union of path and trajectory planning.

1.3.2 FLATNESS-BASED MOTION PLANNING

As defined in the subsection 1.3.1 the motion planning goal is computing a trajectory that satisfies certain path constraints.

Let us define a non linear system $\dot{x} = f(x, u)$. The motion planning consists in fulfill initial and final conditions [40]:

$$\begin{aligned} x(t_i) = x_i,\ u(t_i) = u_i \\ x(t_f) = x_f,\ u(t_f) = u_f \end{aligned} \tag{1.14}$$

Once the path defined, the trajectory generation problem consists in finding a trajectory $t \mapsto (x(t), u(t))$ for $t \in [t_i, t_f]$ that satisfies the system constraints and the initial and final conditions (1.14). Trajectory constraints of type $(x(t), u(t)) \in A(t)$, where $A(t)$ is a submanifold of $X \times U$ could be added to the motion planning initial problem. This results in a growing complexity that requires an iterative solution by numerical methods to find the control input u that satisfies the initial and final conditions (1.14). This iterative process can be solved by using optimal control techniques, however for nonlinear systems some problems still unsolved. Besides this solution needs to integrate the system equations in order to evaluate the solution proposed.

Motion planning by flatness, does not need to integrate the system equations and for a flat output trajectory, command inputs can be computed directly, the u vector resultant always respect the system dynamics, see equation (1.3). By consequence the solutions of the set of differential equations are found. See [44, 59].

Definition 1.1 implies that every system variable can be expressed in terms of the flat outputs and a finite number of its time derivatives. By consequence if we want to compute a trajectory whose initial and final conditions are specified, it suffices to construct a flat output trajectory to obtain the open loop control inputs satisfying the system output desired.

In order to compute all the system variables, the flat output trajectory created needs to be at least r times differentiable, where r is the maximal time derivative of the flat output appearing in the differential flat equations. Additionally this trajectory is not required to satisfy any differential equation. By consequence the flat outputs trajectories can be created by using a simple polynomial approach. See Appendix A for further details.

If the trajectories needs to be optimal in some sense, a more advanced trajectory generation technique has to be used, some application examples can be found in [8, 44, 45, 73].

Let us retake the example 1.2.2 in order to create nominal trajectories for the state θ and the two control inputs u and φ, the desired value for the x and y position is the same and its equal to five, see Fig. 1.3. After computing the time derivatives of such trajectories and using the expression of the equations 2.14, 2.15 and 1.13 it is straightforward to obtain the nominal trajectory for the remaining state and the control inputs. See Figs. 1.3 and 1.4. The fact that obtaining the control inputs is especially helpful to control the system in open loop.

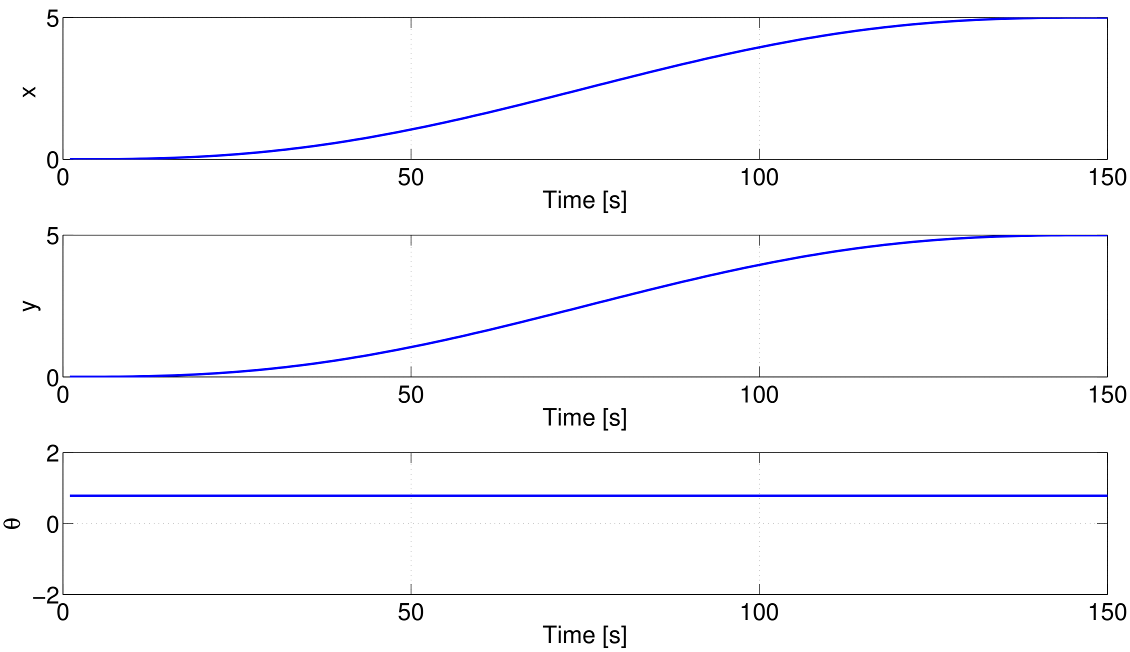

Figure 1.3: Flat outputs and states

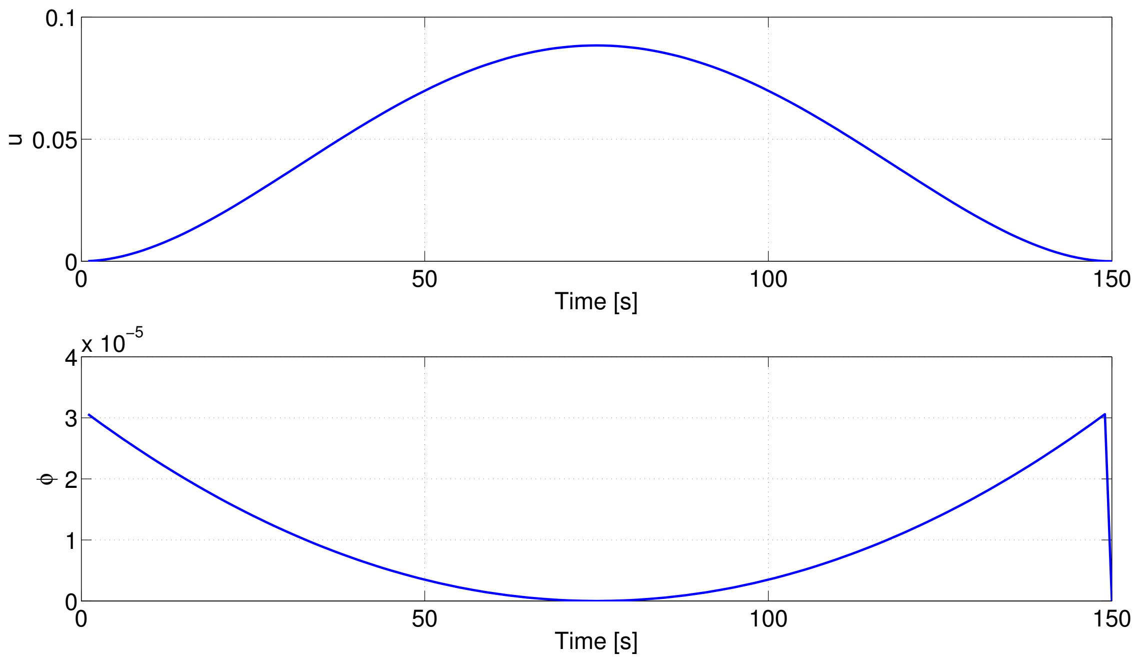

Figure 1.4: Control inputs

1.4 FAULT TOLERANT CONTROL

Industrial and transport systems have become a complex network composed of processors, interfaces, actuators and sensors which may suffer malfunctions. This phenomena could compromise the entire system if a fault occurs. A fault tolerant control (FTC) is designed keeping in mind such potential system components failures, and avoids system loss which can affect productivity or safety as a result.

FTC is divided in two different approaches:

- Passive: Known as robust control, here, the control law is designed to be insensitive to some faults. This approach has limited fault-tolerant capabilities and it is beyond the scope of this work, interested readers are referred to [84] and references therein.

- Active: In this approach, the control system is reconfigured using the information coming from the detection block, having as goal to maintain at least system stability and at the best the nominal behavior. See [58, 65].

This research work is focused on Active Fault Tolerant Control approach.

1.4.1 Active Fault tolerant control systems (AFTCS)

Active approaches consists of adjusting the controller on-line, according to the detected fault, having as goal the preservation of the faulty system performance close to the nominal one, to this a fault recovery task is needed.

For critical failures as an actuator lost for instance, the nominal behavior cannot be maintained, thus, the system performance is reduced as shown in Fig. 1.5. FTC objective is to reconfigure the controller as fast as possible in order to maintain the nominal performance. Moreover, not all faults are reconfigurable, thus is it impossible to keep the system operating even in a degraded mode. In this case FTC function is to shut down the systems safely.

By this way three activities must be covered by the FTC system, [65]:

- Deal with various kinds of faults (sensor, actuators, system itself).
- Provide information about the fault and the reachable performance.
- Decide if the system can still operating or not.

AFTCS overall structure is typically composed by four sub-systems, [83]. see Fig. 1.6.

- A reconfigurable feedforward/feedback controller, which can react to the failure by changing some controller parameters or the entire closed loop.

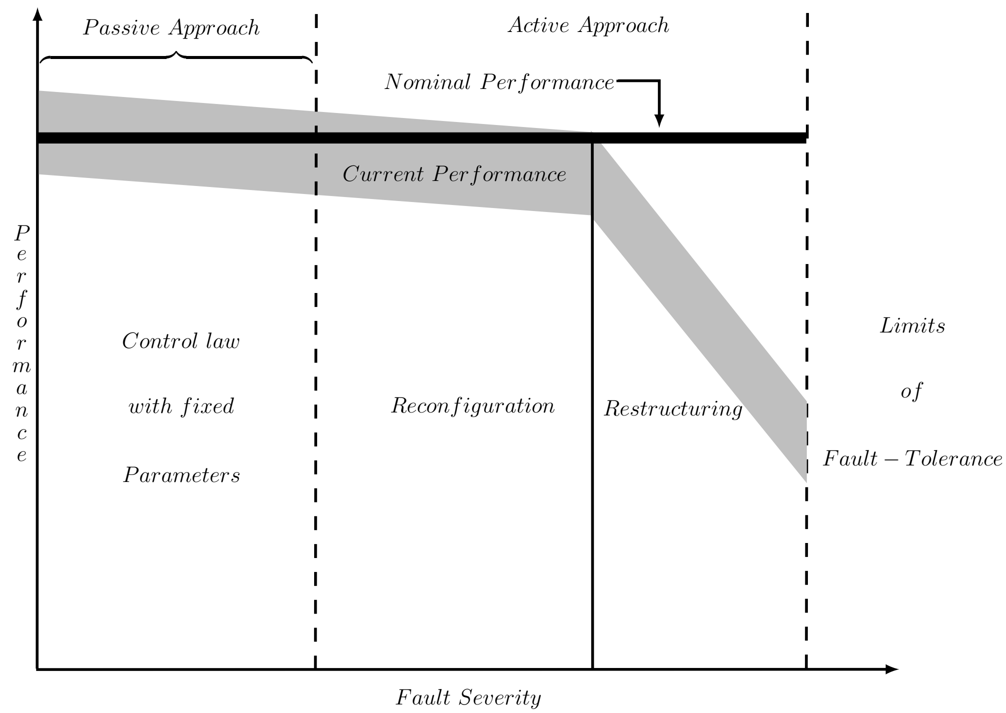

Figure 1.5: FTC Strategies [65]

- A Fault detection and identification (FDI) block, this block has to perform a fast and accurate failure recognition.
- A controller reconfiguration mechanism, which is in charge of link the fault identification mechanism and the reconfigurable controller.
- A trajectory planner/re-planner designed to avoid actuator saturation and adjust the reference trajectory after failure.

Thanks to its versatility, differential flatness can be used to create a completely FTC structure. In fact since its formulation in 1992, flatness has been widely used to design controllers, see [32, 51, 60, 68, 81] for some examples. Regarding fault detection and identification some works has been presented too, [43, 54, 61]. The second of them is issued from the works of this dissertation. Motion planning/replanning can be afforded using flatness, see [8, 9, 45] for some examples. Control reconfiguration has been studied as well. [49, 53, 55, 72]. More details of the FDI and FTC approaches will be presented in the next chapter.

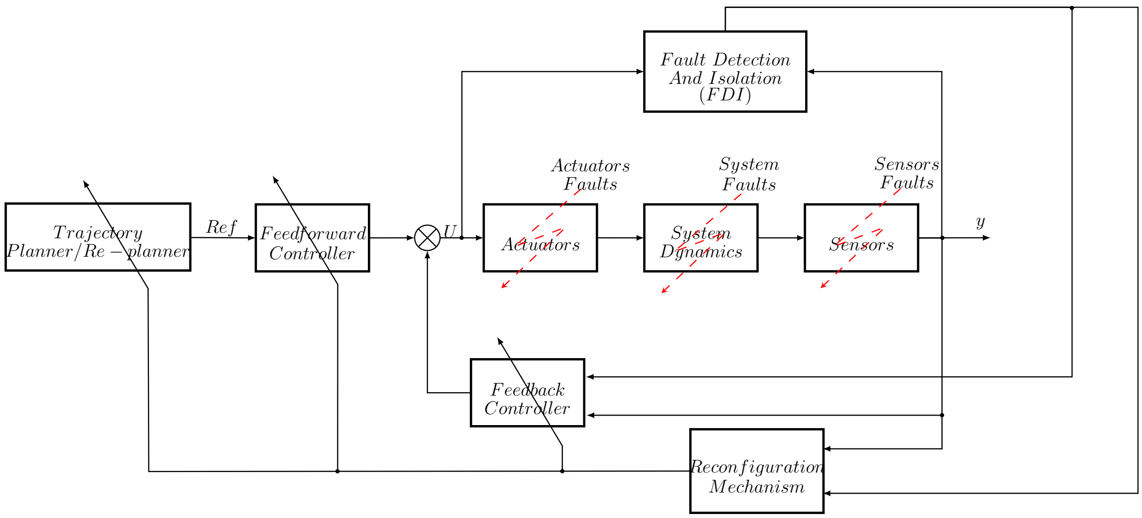

Figure 1.6: General structure of AFTCS [83]

1.5 FAULT DETECTION AND ISOLATION (FDI)

This stage can be divided in two essential tasks [18]:

- Fault detection: in charge of detect the non-expected behavior of the system.
- Fault isolation: localize the faulty system element.

A third activity can be added. Which consists in determine the amplitude of the fault. Along with control reconfiguration FDI block play an important role inside AFTCS. According to [77], FDI methods can be divided in two different groups, see Fig. 1.7:

- Model-based Methods:
    - Quantitative methods: Based on mathematical functional expressing relationships between inputs and outputs of the system, [77].
    - Qualitative methods: Based on qualitative functions expressing the relationships between inputs and outputs, [75].

- Process history based: Based on the availability of large amount of historical process, [76].

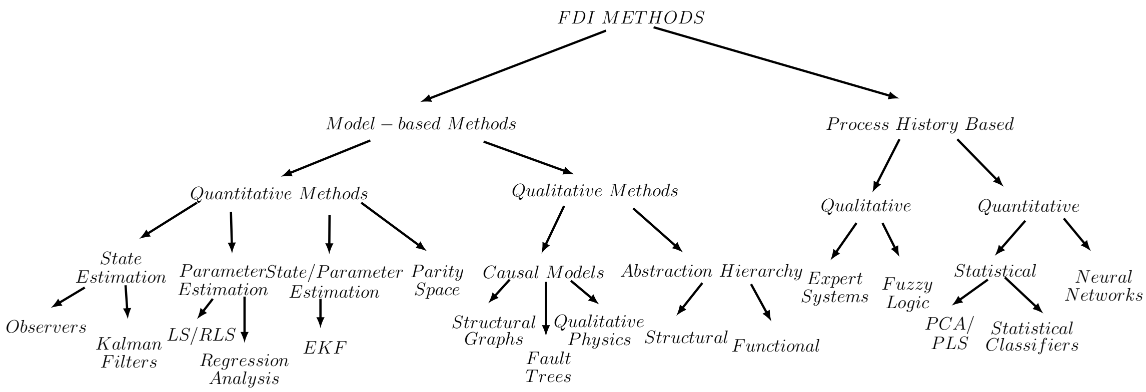

Figure 1.7: FDI Methods [83]

In order to isolate the fault, the three approaches presented below need some a priori knowledge, this is, the set of faults and their relationship with the residues. This information is normally sorted in form of a table. Qualitative model-based and process histories are outside of the boundaries of this thesis, interested reader could find a review of them in [75] and [76] respectively.

1.5.1 QUANTITATIVE MODEL-BASED FDI APPROACH

Quantitative methods need a mathematical model of the system, in order to compute residual signals, which reflects the faults affecting the system. Then, this information is introduced into a decision rule. The union of those tasks helps to obtain information about the fault affecting the plant.

In order to compute the residual signals, redundancy is needed, this could be obtained through two different approaches:

- Hardware Redundancy
- Analytical Redundancy

Figure 1.8 shows the differences between these two approaches.

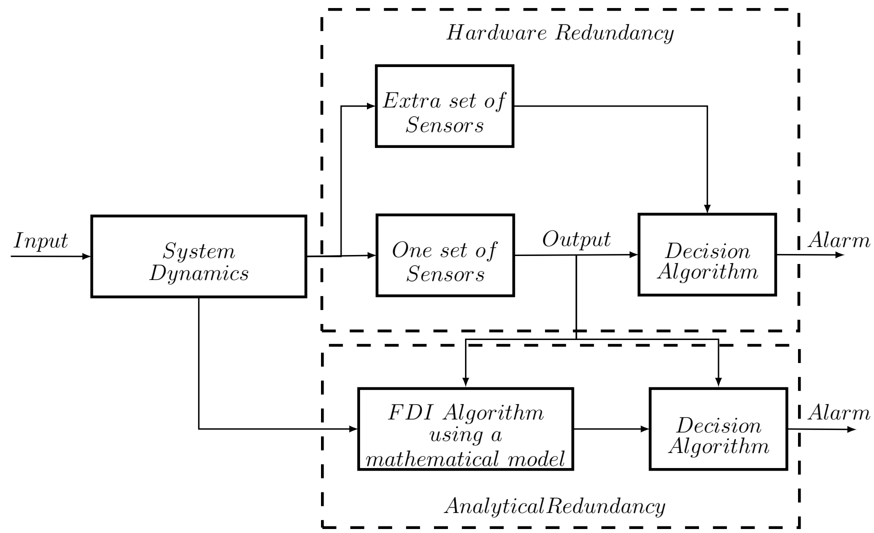

Figure 1.8: Hardware redundancy and analytical redundancy schema

HARDWARE REDUNDANCY

Also known as physical redundancy, it is extremely used in chemical industries, aeronautics and industrial processes where persons life are in danger, the main idea consists in multiply the number of sensors dedicated to perform the same activity. Those sensors usually are based in different technologies.

For instance three sensors, produces three residues as follows:

$$\begin{aligned} r_1 &= m_1 - m_2 \\ r_2 &= m_1 - m_3 \\ r_3 &= m_2 - m_3 \end{aligned} \tag{1.15}$$

A voting mechanism points the faulty sensor by using the table 1.1. In which the symbol $\boxed{\checkmark}$ means that the sensor is fault-free and $\boxed{\times}$ means that the sensor is faulty.

This approach has as advantage an easy design and efficiency. In the other hand since sensors are multiplied the construction and operation costs are elevated.

Sensor 1	Sensor 2	Sensor 3	r_1	r_2	r_3
$\checkmark$	$\checkmark$	$\checkmark$	0	0	0
$\times$	$\checkmark$	$\checkmark$	$\neq 0$	$\neq 0$	0
$\checkmark$	$\times$	$\checkmark$	$\neq 0$	0	$\neq 0$
$\checkmark$	$\checkmark$	$\times$	0	$\neq 0$	$\neq 0$
$\checkmark$	$\times$	$\times$	$\neq 0$	$\neq 0$	$\neq 0$
$\times$	$\checkmark$	$\times$	$\neq 0$	$\neq 0$	$\neq 0$
$\times$	$\times$	$\checkmark$	$\neq 0$	$\neq 0$	$\neq 0$
$\times$	$\times$	$\times$	$\neq 0$	$\neq 0$	$\neq 0$

Table 1.1: Hardware redundancy FDI logic

ANALYTICAL REDUNDANCY

In contrast to physical redundancy, here, the redundant signals are computed via mathematical equations describing the plant.

Residual signals are constructed by making a comparison between the real process and the mathematical model, it is straightforward to think that if the system is unfaulty, the residual signal will be equals to zero, however if a fault affects the plant, the resultant residue will be different to zero. This approach is correct if the mathematical model describes perfectly the process and disturbances are not present. In practice this behavior is not possible, since model uncertainties are always present. Moreover residue post-processing is necessary to distinguish the effects of different faults. After fault generation the principal concern is to obtain the maximum quantity of fault information from them.

Different quantitative model-based FDI techniques for linear and nonlinear systems are presented below.

1.5.1.1 STATE ESTIMATION

The main idea of those techniques is to provide an estimate of the system state $\hat{x}$, from measurements of the control inputs and system measurements. By this way residual signals for sensor faults can be computed by simply comparing the actual output and the estimated one. $r = y - C\hat{x}$ for linear systems and $r = y - H(\hat{x}, u)$ for nonlinear models.

In order to compute the estimated state, different approaches can be used, here a non-exhaustive list of techniques is shown.

1.5.1.1.1 OBSERVER BASED

The observer-based approaches are the mostly applied model-based residual generation techniques. This technique is based in the reconstruction of the outputs of the system, by means of the mathematical model of the plant operating in nominal mode. It should be noted that there is a difference between observers used for control purposes and fault detection. The observers needed for control are state observers, meaning that they estimate states which are not directly measured, with the goal to use such estimations to control the concerned plant. On the other hand, the observers needed for fault detection generate estimation of the measurements, both of them are then compared in order to compute a residual signal. Any deviation of residual signal from zero will trigger a fault alarm. However, the presence of modeling uncertainties and disturbances is inevitable. Therefore, the aim is to design observers such that the effect of the disturbances and uncertainties on the residual signal is reduced while the affect of faults is considerably increased.

Consider the nonlinear system described by the equations

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned} \tag{1.16}$$

where $x(t) \in \Re^n$ is the state vector, $u(t) \in \Re^m$ is the control input, $y(t) \in \Re^p$ is the output of the system.

The observer-based fault diagnosis problem is find a residual generator of the form:

$$\begin{aligned} \dot{\xi}(t) &= g(\xi(t), y(t), u(t)),\ \xi(0) = \xi_0 \\ r(t) &= R(\xi(t), y(t), u(t)) \end{aligned} \tag{1.17}$$

Over the past, several observer-based approaches have been proposed, see for example [11, 18, 24] for a survey. Some of them are presented in the next paragraphs.

NONLINEAR IDENTITY OBSERVER APPROACH (NIO)

Proposed for the first time in [31]. This observer is developed under the assumption that the model is perfectly known and the system is unfaulty. The observer structure is the following:

$$\begin{aligned} \dot{\xi}(t) &= f(\xi, u) + K_{obs}(\xi, u)[y - \hat{y}] \\ r(t) &= y - h(\xi, u) \end{aligned} \tag{1.18}$$

The error estimation can thus be defined by: $e(t) = x(t) - \xi(t)$, its dynamics can be expressed as follows:

$$\begin{aligned} \dot{e} &= F(\xi, u) - K_{obs}(\xi, u)H(\xi, u)e + HOT \\ r(t) &= H(\xi, u)e + HOT \end{aligned} \tag{1.19}$$

Where $F(\xi, u) = \frac{\delta h((x,u))}{\delta x}|_{x=\xi}$ and $H(\xi, u) = \frac{\delta f((x,u))}{\delta x}|_{x=\xi}$. The High Order Terms ($HOT$) are neglected.

The observer gain K_{obs} is determined in such a way that the error dynamics are asymptotically stable $e = 0$. A solution to this problem was first proposed in [1] by assuming that the measurements are linear.

EXTENDED LUENBERGER OBSERVER

The first application of a Luenberger observer was devoted to linear systems [13]. This approach can directly be applied to nonlinear systems, however if the system is operating far away from the linearizing point, the linearized system could deviate largely from the nonlinear model.

The main idea of the extended version of the Luenberger observer is linearize the model around current states estimations ($\hat{x}$), instead of a fix point. Once that a more accurate linearization is computed the observer can be applied. The observer structure is described by:

$$\begin{aligned} \dot{\xi} &= f(\xi, u) + L(\xi, u)(y - C(\xi, u)),\ \xi(0) = \xi_0 \\ \hat{y} &= C(\xi, u) \end{aligned} \tag{1.20}$$

Where $L(\xi, u)$ is the observer gain, which has to assure that the eigenvalues of $\frac{\delta f(x,u)}{\delta x} - L(\xi, u)\frac{\delta c(x,u)}{\delta x}$ are stable. Detailed information and FDI application can be found in [1, 80]. The practical application of this approach is not optimal, since the observer gain has to be computed repetitively, which means an important computational charge.

SLIDING-MODE OBSERVER

Robustness against disturbances and uncertainties are inherent to this kind of observers. Those characteristics make them suitable for state estimation and fault detection. Its design is divided in two stages, first a sliding control surface has to be constructed and then, a control law is designed, which drives the system trajectories to the sliding surface in a finite time. This approach have been widely applied in linear [19, 30] and nonlinear systems [78, 79].

1.5.1.2 PARAMETER ESTIMATION

Parameter estimation approaches are based on the assumption that the fault are reflected in the system parameters. The detection task is accomplished by comparing the nominal parameters versus the on-line estimation. The main advantage of this approach is that it yields the size of the parameter deviation which is important to fault analysis. Parameter estimation is useful for component fault detection since it verifies directly the discrepancy between internal parameters. A disadvantage is that an input signal is always needed in order to excite the system and create signals to estimate the parameters, this action may result in problems if the system is operating in the stationary mode.

Most of the parameter estimation techniques are based on least squares (LS), recursive least squares (RLS), extended least squares (ELS), etc.

1.5.1.3 SIMULTANEOUS STATE/PARAMETER ESTIMATION

1.5.1.3.1 EXTENDED KALMAN FILTER

The extended Kalman filter (EKF) have been largely applied to estimate states and system parameters of discrete systems. Let us consider the discrete nonlinear system described by:

$$\begin{aligned} x_k &= f(x_{k-1}, u_k, v_k, \theta_k) \\ y_k &= h(x_k, w_k, \theta_k) \end{aligned} \tag{1.21}$$

where $x \in \Re^n$ the state vector, $u \in \Re^m$ the control input vector, $y \in \Re^p$ the output measures, $v \in \Re^n$ and $w \in \Re^p$ are the state and measure noise respectively and $\theta \in \Re^q$ is the vector of parameters.

The main idea of the EKF is linearize the nonlinear functions f and h around the current state estimation $\hat{x_k}$, and then the Kalman filter is applied.

The Kalman filter is compound by a group of recurrent equations, which are relatively easy to solve from a numerical point of view. The filter provides the optimal estimation of the states and the variance of the estimation error.

$$\begin{aligned} x_k &= A_k x_{k-1} + B_k u_k + G_k v_k \\ y_k &= c_k x_k + E_k w_k \end{aligned} \tag{1.22}$$

where v and w are non-correlated white noises with zero mean.

$$\begin{aligned} E[v_k v_j^T] &= Q_k \delta kj \\ E[w_k w_j^T] &= R_k \delta kj \\ E[w_k v_j^T] &= 0 \quad \forall k, j \end{aligned} \tag{1.23}$$

where Q and R are the variance matrices of the noise, $E[.]$ is the expectation value of the alleatory variable $[.]$ and $\delta_{kj} = 1, k = j$ and $0, k \neq j$.

The state x and the measure y are deducted from the white noises v and w and the initial condition x_0, with $E[x_0] = 0$. From the initial conditions the covariance matrix $P_0 = E[x_0 x_0^T]$. The goal is provide an estimation of the state vector $\hat{x}_k$, by minimizing the variance of the error estimation.

$$\hat{x}_k = argmin\left\{[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T \mid y_{1:k}]\right\} \tag{1.24}$$

For further details see for example [15, 64].

1.5.1.4 PARITY SPACE

The parity space approach first presented in [12], makes use of the parity check on the consistency of parity equation by using system measures. By this way the inconsistency in the parity relations indicates the presence of fault. Chow and Willsky derived the parity relations based on state-space model of the system, an approach based in transfer functions was developed in [28]. The main idea of this approach is to eliminate the unknown state and then to obtain relations where all components are known.

For nonlinear systems an approach based on the inverse model of input-output is presented in [37, 62] generalized the parity space approach for linear systems to nonlinear systems described by Takagi-Sugeno fuzzy models. [39] presents an extension of the linear parity space approach to nonlinear systems by preserving the original structure of the polynomial parity vector approach. This approach allows to generate the maximum number of linearly independent residues for a system. Detailed information of the method can be found in chapter 2.

1.6 FAULT RECOVERY

After the fault isolation stage, the next step in a FTC system is re-adjust the control chain of the system, aiming the nominal behavior or at least stability. This task can be accomplished in two different ways [47, 67].

- Fault accommodation.
- Fault reconfiguration.

Figure 1.9 and the next subsections presents a general overview of both group of methods.

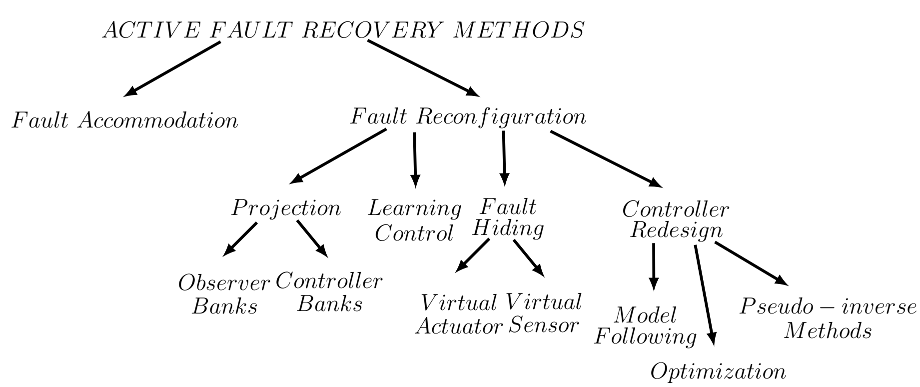

Figure 1.9: Fault recovery methods [47]

1.6.1 Fault Accommodation

Here, the measurements and the signals going into the controller remain unchanged, the fault recovery is carried out by changing the controller internals (dynamic order, parameters, gain values etc.), [58]. One example of this technique is the adaptive controller technique, where the controller is tuned to minimize the distance between nominal closed loop and the actual behavior, [4].

Adaptive controller can be divided in two approaches: direct and indirect, in the first one the controller parameters are directly tuned, the second is performed in two steps, first, the mathematical model of the plant is estimated and then a controller for this plant is computed. This approach presents some limitations, for example, if an abrupt fault affects the system, the needed time to compute all the controllers parameters could be important. By consequence the system could become unstable before finishing the compute. The same case is worst with the second approach, since more time is needed to carried out the two steps. Besides, the fault can lead the system outside the linearization zone and by consequence linearize the system becomes impossible. Structural damage is not covered.

1.6.2 Fault reconfiguration

In fault reconfiguration techniques controller parameters and input-output signals are manipulated. By this way those techniques carry out fault recovery not only reconfiguring the controller but also including dynamic signal re-routing of measures.

Reconfiguration methods are divided in four different groups, [47]:

- Projection.
- Controller redesign.
- Fault hiding.

- Learning control.

1.6.2.1 PROJECTION

Methods inside this classification are always based in the off-line design of certain components. Those elements are arranged in banks. Depending on the fault that want to be reconfigured, two banks can be used, see Fig. 1.10:

1.6.2.1.1 BANK OF OBSERVERS

Observers banks can only handle sensor faults, each observer uses the information of all sensors but one. This measure is considered hypothetically faulty. Using all inputs and all outputs except the one considered faulty, each observer can compute an estimation of every system state, and thus estimate the plant output $\hat{y}$. Residues are obtained by computing the difference between the measure y and the estimation $\hat{y}$. The residue with the smallest error represents the current fault case, [25]. Reconfiguration is achieved by feeding the nominal controller with the current fault case. The main advantage of this technique is that it handle the reconfiguration in an integrated manner, which by consequence can reduce the computing time.

1.6.2.1.2 BANK OF CONTROLLERS

Sensors, actuators and system faults can be covered with this technique. The FDI is carried out by a diagnostic algorithm. The fault information coming from this block is then used to select the most appropriate of the a priori designed controllers. Since, the number of controllers designed must be equal to the number of managed failures the off-line effort can be important.

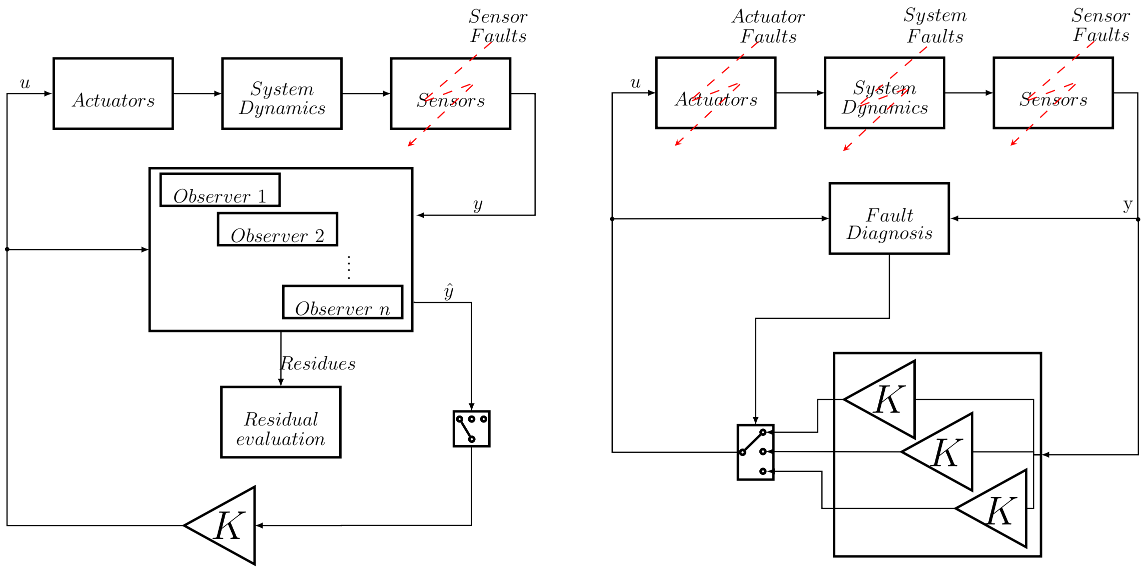

Figure 1.10: Banks of observers and controllers [47]

1.6.2.2 CONTROLLER REDESIGN

This approach perform in real time a completely redesign of the controller in order to recover the faulty system. This action is carried out in explicit or implicit way, in the first, the difference between the outputs of the reconfigured plant and the reference model is minimized. In the implicit way, quadratic functions of the actual and modeled states are minimized. Computational cost varies from one method to another. The goal of the control reconfiguration is to minimize the distance between a nominal unfaulty model and the faulty system. This problem is known as model matching. Mathematically, this idea can be expressed as follows:

Let us define a linear system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_d d_x(t) \\ y(t) &= Cx(t) + d_y \end{aligned} \tag{1.25}$$

Where $x(t) \in \Re^n$ is the system state, $y(t) \in \Re^p$ is the output, $u(t) \in \Re^m$ the control input, B_d the disturbance distribution matrix, d_x and d_y are the state and measurements disturbance respectively.

In presence of faults the system (1.25) becomes:

$$\begin{aligned} \dot{x}_f(t) &= A_f x_f(t) + B_f u_f(t) + B_d d_x(t) \\ y(t) &= C_f x_f(t) + d_y \end{aligned} \tag{1.26}$$

The idea of model matching is define a reference model compound by the system (1.26) and the state feedback controller:

$$u(t) = Kx(t) + Gw(t) \tag{1.27}$$

Where K is the static controller feedback matrix, G is the reference pre-filter, $w(t)$ the reference input. The reference model is expressed as follows:

$$\begin{aligned} \dot{x}(t) &= Mx(t) + Nw(t) \\ y &= P^* x \end{aligned} \tag{1.28}$$

In transfer function form, the reference model is:

$$T(s) = P^* \left(sI - M\right)^{-1} N \tag{1.29}$$

Where $M = A - BK$ and $N = BG$. M, N and P^* are selected by the designer. Thus the model matching problem consist in determine a new feedback controller:

$$u(t) = K_f x(t) + G_f w(t) \tag{1.30}$$

such that:

$$\begin{aligned} A_f - B_f K_f - M = 0 \\ B_f G_f - N = 0 \end{aligned} \tag{1.31}$$

Various approaches have been developed to solve this problem.

1.6.2.2.1 PSEUDO-INVERSE METHODS

This method was the first one to treat this problem [5]. It is addressed to actuators and fault systems. Here the model matching problem presented above is solved by minimizing the distance between the closed loop matrices according to the 2-norm $\parallel . \parallel_2$. Two criteria are minimized:

$$\begin{aligned} J_1 &= \parallel M - (A_f - B_f K_f) \parallel_2 \\ J_2 &= \parallel N - B_f G_f \parallel_2 \end{aligned} \tag{1.32}$$

The optimal solution can be computed by using:

$$\begin{aligned} K_f^* &= argmin J_1 = B_f^+ (A_f - M) \\ G_f^* &= argmin J_2 = B_f^+ N \end{aligned} \tag{1.33}$$

Where B_f^+ denotes the pseudo-inverse of B_f. The optimal solution obtained (K_f^*, G_f^*) is plugged into the loop instead of the nominal controller, see Fig1.11. This method does not guarantee the stability of the reconfigured system because the optimisation problem is unconstrained. In order to ensure system stability Gao and Antsaklis presented a modified pseudo-inverse method (MPIM) [26]. Here the unconstrained stability problem become a constrained one, it is formulated in terms of the stability robustness of linear systems with structured uncertainty. The stability problem is solved, however the high computational charge prevents its application in real time. In [69] Staroswiecki presented a computationally simpler approach based on a set of admissible models. The admissible model is chosen in such a way that the system robust stability is assured. This technique is known as Admissible pseudo-inverse method (APIM).

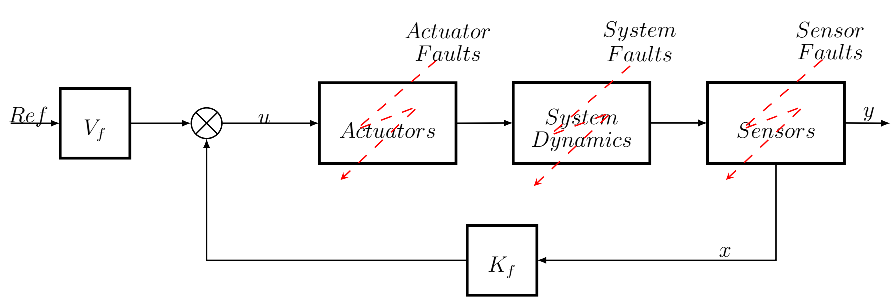

Figure 1.11: Pseudo-inverse Method [47]

1.6.2.2.2 MODEL FOLLOWING

PERFECT MODEL FOLLOWING

This idea was presented in [27]. Here the model matching problem is solved by combining the use of a stabilizing feedback and a dynamic compensator in order to match exactly the dynamic behavior.

A closed loop linear system satisfies perfect model following with respect to the reference model (1.28) if and only if:

$$\begin{aligned} A + BK &= M \\ G &= N \end{aligned} \tag{1.34}$$

Figure 1.12 shows the typical structure. The reference model is running in parallel with the plant and it is implemented in the controller. By this way the control input is:

$$u(t) = K_e e(t) + (K_m x_m(t) + K_w w(t)) \tag{1.35}$$

With K_e the stabilizing gain and K_m, K_w the model matching gains. $e(t)$ is defined as the difference between the state variables of the plant and the reference model, to achieve perfect model matching this error has to be equal to zero for all $t > 0$.

The model matching gains are determined to minimize:

$$\| (M - A_f)x(t) + Nw(t) - B_f u(t) \|_2 = \| \dot{e}(t) - (A_f - B_f K_e)e(t) \|_2 \tag{1.36}$$

Where the error dynamics is expressed by:

$$\dot{e}(t) A_f e(t) + (M - A_f)x_m(t) + Nw(t) - B_f u(t) \tag{1.37}$$

The solution of the equation (1.36) is given by:

$$\begin{aligned} K_m &= B_f^+(M - A_f) \\ K_w &= B_f^+ N \end{aligned} \tag{1.38}$$

This technique guarantees closed loop stability if the terms (A_f,B_f) are stabilisable.

1.6.2.2.3 OPTIMISATION

LQ REDESIGN

This technique was presented in [42].The main idea of this technique is design a LQ-optimal nominal controller. After FDI a new LQ controller is designed online, using the faulty plant model. If the faulty plant still controllable the LQ algorithm will find a new LQ-optimal controller.

MODEL PREDICTIVE CONTROL (MPC)

The main idea of the MPC technique is divided in three main steps, first a prediction of the future behavior of the process state/output is accomplished. Second, the future input signals are computed online at each step by minimizing a cost function under inequality constraints on the manipulated (control) and/or controlled variables. Finally, apply on the controlled plant only the first of vector control variable and repeat the previous step with new measured input/state/output variables.

To achieve control reconfiguration, it is necessary to update the internal plant model of the MPC controller. The MPC controller will find the optimal sequence using the update plant model. Since the computational charge is important, this technique is applied principally in slow dynamics systems.

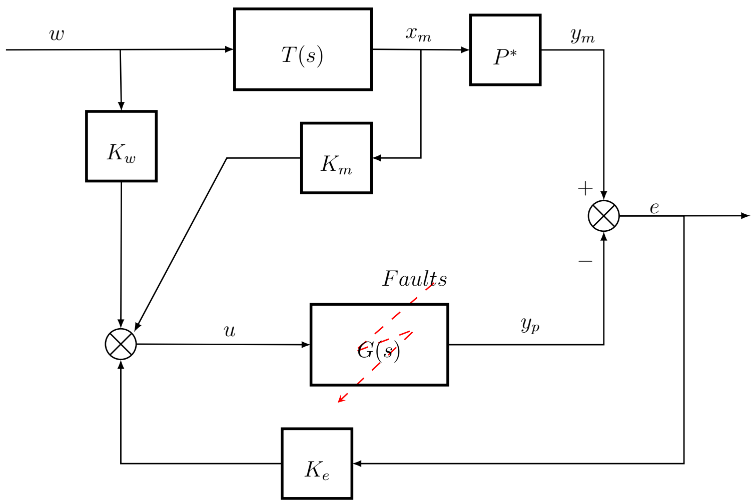

Figure 1.12: Model Following schema [47]

1.6.2.3 FAULT-HIDING

The main idea of these approaches is to keep the nominal controller during a fault occurrence. Fault recovery is then carried out by adding a reconfiguration block between the controller and the faulty plant. This block is designed in a manner that the faulty plant mimics the behavior of the unfaulty system. This behavior is obtained by means of two blocks, a virtual sensor and a virtual actuator Fig. 1.13. A virtual sensor consists in a model of the faulty plant and a gain L, this block is in charge of provide estimates of the system states $\hat{x}$. The virtual actuator is compound of a reference model, as well as feedback of the difference between the state of the reference model and two matrix M and N, which have to be chosen in such a way that the virtual actuator state is stable and the difference between the nominal output and the real one equals to 0.

By this way sensor and actuator faults can be handle, this approach can be applied to linear [47] and nonlinear [66] systems.

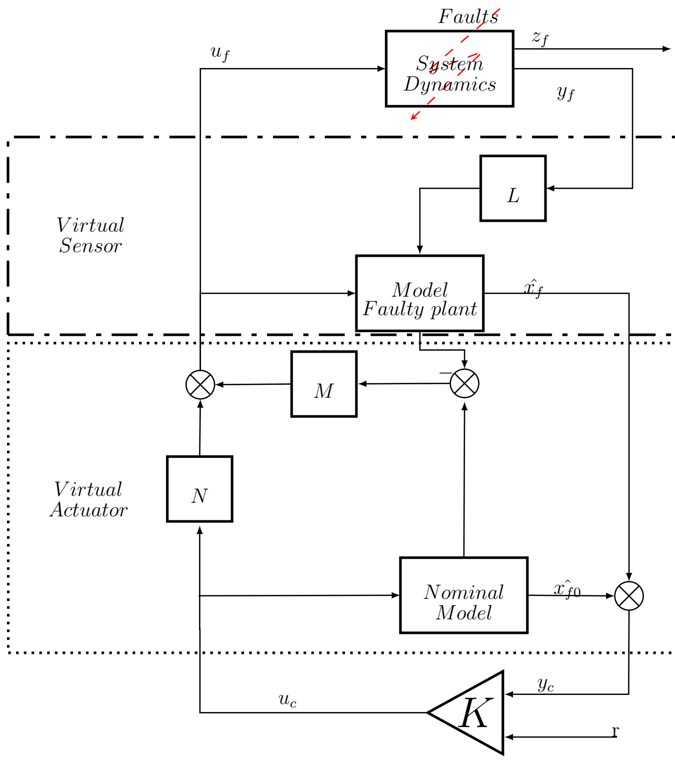

Figure 1.13: Fault-hiding approach [47]

1.6.2.4 LEARNING CONTROL

This approach is based in the idea of mix classical control techniques with learning control methods (Neural networks, expert systems, etc.). By this way a database of performance measures is constructed and a decision making unit decides how to face the fault, [70].

1.7 CONCLUSION

This chapter presented an overview of some model-based FDI techniques and fault reconfiguration approaches. The concept of differentially flat systems are presented as well. Flat systems property possess an inherent capability of generate analytical redundancy, this property can be exploited to create FDI schemes. The two final sections were devoted to present the existent FDI and FTC techniques presented currently in the literature.

In the next chapter the FTC flatness-based proposed approach is detailed. This approach differs from the presented here in the fact that both, FDI and fault recovery are carried out by exploiting the inherent properties of the flat systems.

CHAPTER 2

FAULT TOLERANT CONTROL: A FLATNESS-BASED APPROACH

Abstract:

In this chapter the proposed FTC approach is presented. It is based in the fact that the set of flat outputs is not unique, in fact if a second set of flat outputs algebraically independent of the first one is found, this will provide redundancy, which, will increase the number of residues, facilitating in this manner the fault detection. Additionally the redundant signals will be used to reconfigure the system after fault.

2.1 INTRODUCTION

This chapter presents the proposed approach of this research work.

Chapter 2 is divided in two main parts, in the first one a state of the art of the FDI/FTC flatness-based techniques is presented, together with the mathematical theory that helps to found the flat outputs. The necessary conditions to use the proposed approach are defined as well. In the second part of the section, the FTC flatness-based approach is presented, firstly the attention is only focus in the FDI technique and then using the inherent characteristics of the flat systems, the FDI technique is extended to reconfigure the faulty system. By this way the fault detection and fault reconfiguration tasks are done in an integrated manner.

In order to verify the behavior of the system and decide if a fault is present or not, it becomes necessary to generate residual signals. The proposed approach use parity equations to generate them. Thanks to its inherent properties differential flatness can be used to generate redundancy in a natural way, this property will be exploited to generate parity equations, those equations are a comparison between the behavior of the system and the behavior of a flat model of the fault-free case, the resultant will be the residual signal, which, will be different of zero in presence of a fault and close to zero in the fault-free case.

The differential flatness property is already proven for many systems [40, 52], however nowadays we cannot find an effective and systematic algorithm to compute the flat outputs. In fact founding a set of flat outputs is more related to the experience and sometimes the knowledge of the system because the flat outputs could be a certain physical meaning. Which is already established are the necessary and sufficient conditions for differential flatness, those conditions were presented by Jean Lévine in [41] , this work inspire the FTC based approach presented in this work. This method will be presented in section 2.4. The next two sections are consecrated to present the sate of the art of FDI/FTC flatnes-based approaches.

2.2 FAULT DETECTION AND ISOLATION BY FLATNESS

Since the first publication of flat systems theory in the early 90's, they have been attracted much attention in different automatic control areas, such as controller design [32, 51, 60, 68, 81] and motion planning [8, 9]. However regarding FDI/FTC not many works are published. See for instance [35, 43, 49, 61, 72, 82]. The main property of the flat systems provide analytical redundancy, since, every control input and system state can be expressed as function of flat outputs, see definition 1.1. By this way parity equation could be computed by simply comparing measures versus estimations. The result of such parity equations are known as residual signals, as always if its amplitude is close to zero the system is working normally, if not, the system is consider faulty. Almost all the approaches presented in the literature take advantage of this property.

In [61] differential flatness is coupled with a nonlinear observer, in order to construct the residual signals, see Fig. 2.1. The nonlinear observer is in charge of create an estimation of the control inputs. Such estimation can now be directly compared to the estimation of the same variables but this time obtained with the differentially flat equations. The main problem of this technique lies in the fact that the residuals are obtained by comparing two different approaches, since both of them could differ in some aspects, for instance dynamic speed, this could create some false alarms because the phase difference of both signals.

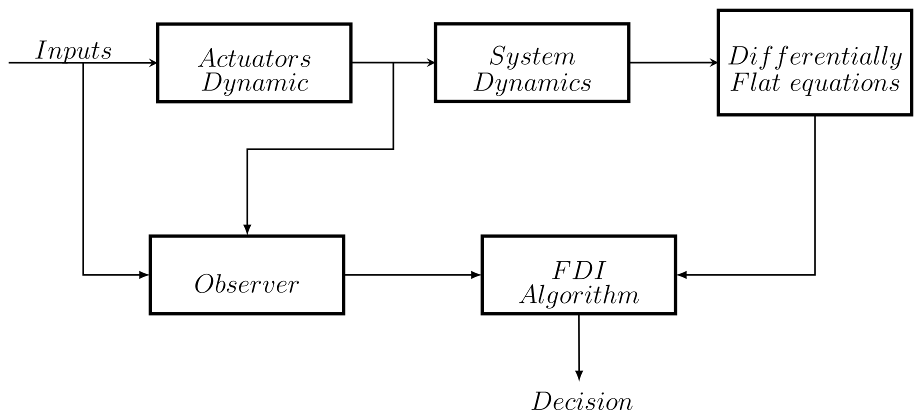

Figure 2.1: FDI flatness-based schema with observer, [61]

The flatness-based FDI approach presented in [48, 49], use an algebraic approach [21, 22, 56] to estimate actuator faults. Such estimations help to identify the fault. This work takes into account only additive faults. The fact that the fault is estimated will be specially useful to reject the fault.

The analytical redundancy obtained thanks to the main property of the differentially flat systems is exploited to generate residual signals in [72]. The time derivatives of the flat output are computed by using B-splines. For this work the fault amplitude is estimated and this information is used to compensate the fault. This approach is applied to linear systems.

In [43] and [82] the residuals are computed by comparing the state estimation to the state measure, in order to overcome noise and modeling errors and improve the effectiveness of the threshold-based fault detection scheme, a probabilistic distribution is generated. This approach is applied to discrete nonlinear flat systems. The main problem of this approach is the fact that create an online probabilistic distribution is a hard task, the response of the authors was coupled a simplified pre-computed distribution with a neuro-fuzzy logic, which reduce the computational charge but increase the designing work. Those approaches are applied to discrete nonlinear flat systems.

In [35] the residual signals are computed by using the estimation of derivatives obtained with the algebraic approach presented in [21, 22, 56]. Besides fault indicators are robust with respect to uncertain parameters in the controlled plant.

Table 2.1 shows a summary of the FDI techniques found on the literature.

Reference	Applied in	Handling faults	Fault type	Advantages/ Inconvenients
[61]	Nonlinear	Sensors	Multiplicative	Relatively easy design. Difference between dynamics could create false alarms.
[48, 49]	Linear Nonlinear	Actuators	Additive	Estimate the fault amplitude. Only additive faults are taken into account.
[72]	Linear	Sensors	Additive	Estimate the fault amplitude. Only additive faults are taken into account.
[43, 82]	Discrete nonlinear	System	Additive	Could facilitate the real time application. Hard design work.
[35]	Nonlinear	Actuators	Additive	Estimate the fault amplitude. Could be computational expensive.

Table 2.1: FDI by flatness

2.3 FAULT TOLERANT CONTROL BY FLATNESS

Recovery after fault is carried out using the estimation of the faults. In fact for instance [72], estimates the fault amplitude by adding the nominal value to the estimated, see Fig. 2.2. Then the signal is conditioned using the B-spline, the obtained trajectory is added to the measure of the faulty sensor. Only faults affecting sensor are taken into account, the approach is applied to linear systems, focus on sensor faults.

As in the reference above in [49] the fault is estimated and then such information helps to recover the system from a faulty position. The main difference lies in the fact that this time the fault is estimated by using the algebraic approach and not B-splines. The approach is intended to actuator faults. According to the authors additive and multiplicative could be treated indistinctly.

The main disadvantage of both techniques is the fact that estimation plus signal conditioning could take some time to be accomplished, such time delay could lead the system to instability.

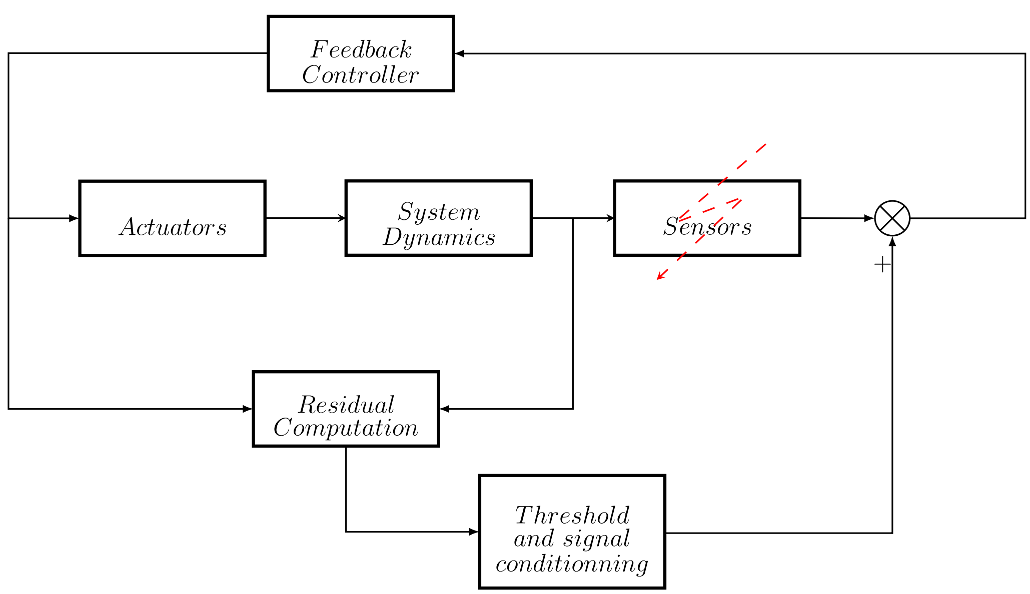

Figure 2.2: FDI flatness-based FTC schema, [72]

2.4 ON THE FORMAL CALCULUS OF FLAT OUTPUTS

More details of the theory presented in this section can be found in [41].

Let us consider a nonlinear system in its implicit form (where the input variables are eliminated).

$$F(x, \dot{x}) = 0 \tag{2.1}$$

An implicit system is defined as $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$, with $\mathfrak{X} = X \times \mathbb{R}_{\infty}^{n}$, $dimX = n$, $\tau_{\mathfrak{X}}$ the trivial Cartan field on $\mathfrak{X}$, and rank $\frac{\delta F}{\delta \dot{x}} = n - m$, this system will be Lie-Bäcklund equivalent to the implicit system $(\mathfrak{Y}, \tau_{\mathfrak{Y}}, G)$, with $\mathfrak{Y} = Y \times \mathbb{R}_{\infty}^{p}$, $dimY = p$, $\tau_{\mathfrak{Y}}$ the trivial Cartan field on $\mathfrak{Y}$, and rank $\frac{\delta G}{\delta \dot{y}} = p - q$ if and only if it exists a locally C^{∞} mapping $\Phi : \mathfrak{Y} \mapsto \mathfrak{X}$, with locally C^{∞} inverse Ψ. This implies that:

- $\Phi_{*}\tau_{\mathfrak{Y}} = \tau_{\mathfrak{X}}$ and $\Psi_{*}\tau_{\mathfrak{X}} = \tau\mathfrak{Y}$.
- For every $\bar{y}$ such that $L_{\tau_{\mathfrak{Y}}}^{k} G(\bar{y}) = 0,\ \forall k \geq 0$, then $\bar{x} = \Phi(\bar{y})$ satisfies $L_{\tau_{\mathfrak{X}}}^{k} F(\bar{x}) = 0,\ \forall k \geq 0$ and conversely.

Where L_{τ_n} is the lie derivative along the Cartan field τ_n.

Definition 2.1 *The implicit system $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$ is flat if and only if it is Lie-Bäcklund equivalent to the trivial system $(\mathbb{R}, \tau_m, 0)$.*

Theorem 2.2 *The system $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$ is flat, if, and only if there exists a locally C^{∞} and invertible mapping $\Phi : \mathbb{R}_{\infty}^{m} \mapsto \mathfrak{X}$ such that:*

$$\Phi^{*} dF = 0 \tag{2.2}$$

Defining the polynomial matrices as follows:

$$dF = \frac{\partial F}{\partial x} dx + \frac{\partial F}{\partial \dot{x}} d\dot{x} = \left(\frac{\partial F}{\partial x} + \frac{\partial F}{\partial \dot{x}} \frac{d}{dt} \right) \triangleq P(F) dx \tag{2.3}$$

$$P(\Phi_0) \triangleq \sum_{j \geq 0} \frac{\partial \Phi_0}{\partial y^{(j)}} \frac{d^j}{dt^j} \tag{2.4}$$

We thus can write

$$\Phi^* dF = P(F)P(\Phi_0)dy \tag{2.5}$$

By consequence, we have to find a polynomial matrix $P(\Phi_0)$ solution to

$$P(F)P(\Phi_0) = 0 \tag{2.6}$$

If F is restricted to be a meromorphic function [1], $P(\Phi_0)$ may be obtained via the Smith decomposition (see Appendix B) of $P(F)$.

The variational system $P(F)$ could be decomposed using the Smith decomposition in:

$$VP(F)U = (I_{n-m}, 0_{n-m,m}) \tag{2.7}$$

Let us define $\mathfrak{K}$ as a field of meromorphic functions from $\mathfrak{X}$ to $\mathbb{R}$, $\mathfrak{K}\left[\frac{d}{dt}\right]$ as the principal ring of $\mathfrak{K}$-polynomials of $\frac{d}{dt} = L_{\tau_{\mathfrak{X}}}$, $\mathcal{M}_{p,q}\left[\frac{d}{dt}\right]$ the module of the $p \times q$ matrices over $\mathfrak{K}\left[\frac{d}{dt}\right]$, with p and q arbitrary integers, and, $\mathcal{U}\left[\frac{d}{dt}\right]$ is the group of unimodular matrices of $\mathcal{M}_{p,p}\left[\frac{d}{dt}\right]$.

By using this notation the set of hyper-regular matrices $P(\Phi) \in \mathcal{M}_{n,m}\left(\frac{d}{dt}\right)$ satisfying (2.2) is given by:

$$P(\Phi) = U \begin{pmatrix} 0_{n-m,m} \\ I_m \end{pmatrix} W \tag{2.8}$$

Where $U \in R - Smith(P(F))$ and $W \in \mathcal{U}_m\left(\frac{d}{dt}\right)$ is an arbitrary unimodular matrix.

Let us define:

$$\hat{U} = U \begin{pmatrix} 0_{n-m,m} \\ I_m \end{pmatrix} \tag{2.9}$$

Lemma 2.3 *For every matrix $Q \in L - Smith(\hat{U})$, it exists a matrix $Z \in \mathcal{U}_m\left(\frac{d}{dt}\right)$ such that:*

[1] A meromorphic function on an open subset D of the complex plane is a function that is infinitely differentiable and equal to its own Taylor series on all D except a set of isolated points, which are poles for the function.

$$QP(\Phi) = \begin{pmatrix} I_m \\ 0_{n-m,m} \end{pmatrix} Z \tag{2.10}$$

Moreover, for every Q, the sub-matrix $\hat{Q} = (0_{n-m,m},\ I_{n-m})\, Q$ is equivalent to $P(F)$.

A flat output of the variational system is given by:

$$w(\bar{x}) = \begin{pmatrix} w_1(\bar{x}) \\ \vdots \\ w_m(\bar{x}) \end{pmatrix} = (I_m,\ )0_{m-n-m})\, Q\bar{x}dx_{|\mathcal{X}0} \tag{2.11}$$

if $dw = 0$, a flat output of the nonlinear implicit system (2.1) can be obtained by integrating the equation $dy = w$. Otherwise, it is necessary to find and integral base, if such base exists. This means that we have to find an integral factor $M \in \mathcal{U}_m\left(\frac{d}{dt}\right)$ verifying $d(Mw) = 0$.

Definition 2.4 *Strongly closed:*

The $\mathfrak{K}\left(\frac{d}{dt}\right)$-ideal Ω, finitely generated by the 1-forms ($w_1, ...w_m$) defined by (2.11), is strongly closed in $\mathcal{X}_0$, (or equivalently, the system $(\mathfrak{X},\ \tau_{mathfrakX}, F)$ is flat) if and only if it exists an operator $\mu \in \mathcal{L}_1((\Lambda(\mathfrak{X}))^m)$, and a matrix $M \in \mathcal{U}_m\left(\frac{d}{dt}\right)$ such that:

$$dw = \mu w, \qquad \mathfrak{d}(\mu) = \mu^2, \qquad \mathfrak{d}(M) = -M\mu \tag{2.12}$$

Where $\mathcal{L}_1((\Lambda(\mathfrak{X}))^m)$ is the space of linear operators which maps the p-forms of dimension m in $\mathfrak{X}$ in (P+1)-forms of dimension m in $\mathfrak{X}$, $\mathfrak{d}$ represents the extension of the exterior derivative d, where the coefficients have their value in $\mathfrak{K}\left(\frac{d}{dt}\right)$.

Additionally if the relation (2.12) is satisfied, a flat output z can be obtained by integrating the system of equations $dy = Mw$.

This can be resumed in the next algorithm:

- Compute the variational system $P(F) = \frac{\delta F}{\delta x} + \frac{\delta F}{\delta \dot{x}}\frac{d}{dt}$ of (2.1), if $P(F)$ is not hyper-regular the system is not flat.

- Compute the smith decomposition of $P(F)$.
- Compute $\hat{U} = U \begin{bmatrix} 0_{n-m,m} \\ I_m \end{bmatrix}$.
- Obtain the Smith decomposition of $\hat{U}$.
- Compute the vector of 1-form ω defined in (2.11)
- Obtain the operator μ, such that $d\omega = \mu\omega$ by identification term by term, if possible.
- If not, among the possible operators μ, keep only the operators who verifies that $\mathfrak{d}(\mu) = \mu^2$.
- Determine by identification term by term, a matrix M, which validates $\mathfrak{d}(M) = -M\mu$.
- Between all the options of matrix M, keep only the unimodular matrices, if such matrix does not exists, the system is not flat. On the contrary, a flat output can be obtained by integrating the system of equations $dy = M\omega$.

Example 2.5 *Non holonomic car, see example 1.2.2*

The system equations of the non holonomic car in its implicit representation are:

$$F(x, y, \theta, \dot{x}, \dot{y}, \dot{\theta}) = \dot{x}sin\theta - \dot{y}cos\theta = 0 \tag{2.13}$$

The first step is to compute the variational system:

$$\begin{aligned} P(F) &= \left(\frac{\partial F}{\partial x} + \frac{\partial F}{\partial \dot{x}} \frac{d}{dt},\ \frac{\partial F}{\partial y} + \frac{\partial F}{\partial \dot{y}} \frac{d}{dt},\ \frac{\partial F}{\partial \theta} + \frac{\partial F}{\partial \dot{\theta}} \frac{d}{dt} \right) \\ &= \left(sin\theta \frac{d}{dt},\ -cos\theta \frac{d}{dt}\ \dot{x}cos\theta + \dot{y}sin\theta \right) \end{aligned} \tag{2.14}$$

Defining $E = \dot{x}cos\theta + \dot{y}sin\theta$, and permuting columns we can write the variational system as follows:

$$P(F) = \left(E,\ -cos\theta \frac{d}{dt},\ sin\theta \frac{d}{dt} \right) \tag{2.15}$$

After applying the Smith decomposition algorithm, we obtain the unimodular matrix U:

$$U = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ \frac{1}{E} & \frac{cos\theta}{E}\frac{d}{dt} & -\frac{sin\theta}{E}\frac{d}{dt} \end{bmatrix} \tag{2.16}$$

Defining $\hat{U}$ as:

$$\hat{U} = U\,(0_{1,2},\ I_2)^T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \frac{cos\theta}{E}\frac{d}{dt} & -\frac{sin\theta}{E}\frac{d}{dt} \end{bmatrix} \tag{2.17}$$

After computing the Smith decomposition of $\hat{U}$ we obtain:

$$Q = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ \frac{sin\theta}{E}\frac{d}{dt} & -\frac{cos\theta}{E}\frac{d}{dt} & 1 \end{bmatrix} \tag{2.18}$$

Multiply the matrix Q by $(dx,\ dy,\ d\theta)^T$. The last line is equal to $\frac{1}{E}(sin\theta d\dot{x} - cos\theta d\dot{y} + (\dot{x}cos\theta + \dot{y}sin\theta)d\theta) = \frac{1}{E}d(\dot{x}sin\theta - \dot{y}cos\theta)$, which is equal to zero, see (2.14). The remaining part of the system:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} dx \\ dy \\ d\theta \end{bmatrix} = \begin{bmatrix} w1 \\ w2 \end{bmatrix} \tag{2.19}$$

is trivially strongly closed with $M = I_2$, which finally gives the same set of flat outputs $z = [y,\ x]^T$computed in example 1.2.2.

The set of flat outputs is not unique, since it depends on the decomposition of $P(F)$ and how explained in [41] it is not unique neither. Let us illustrate this with the next example.

Example 2.6 *Non holonomic car (2)*

By right multiplying the variational system depicted in (2.15) by:

$$Q = \begin{bmatrix} cos\theta & 0 & 0 \\ sin\theta & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.20}$$

an using $sin\theta \frac{d}{dt}(cos\theta) - cos\theta \frac{d}{dt}(sin\theta) = -\dot{\theta}$. The Smith decomposition of $P(F)$ is given by:

$$U = \begin{bmatrix} cos\theta & -\frac{1}{\theta}cos^2\theta\frac{d}{dt} & \frac{1}{\theta}(\dot{x}cos\theta + \dot{y}sin\theta)cos\theta \\ sin\theta & 1 - \frac{1}{\theta}sin\theta cos\theta\frac{d}{dt} & \frac{1}{\theta}(\dot{x}cos\theta + \dot{y}sin\theta)sin\theta \\ 0 & 0 & 1 \end{bmatrix} \tag{2.21}$$

The matrix $\hat{U}$ is equal to:

$$\hat{U} = \begin{bmatrix} -\frac{1}{\theta}cos^2\theta\frac{d}{dt} & \frac{1}{\theta}(\dot{x}cos\theta + \dot{y}sin\theta)cos\theta \\ 1 - \frac{1}{\theta}sin\theta cos\theta\frac{d}{dt} & \frac{1}{\theta}(\dot{x}cos\theta + \dot{y}sin\theta)sin\theta \\ 0 & 1 \end{bmatrix} \tag{2.22}$$

The Smith decomposition of $\hat{U}$ gives:

$$Q = \begin{bmatrix} -tan\theta & 1 & 0 \\ 0 & 0 & 1 \\ \frac{1}{\theta}sin\theta cos\theta\frac{d}{dt} & -\frac{1}{\theta}cos^2\theta\frac{d}{dt} & -\frac{1}{\theta}(\dot{x}cos\theta + \dot{y}sin\theta)cos\theta \end{bmatrix} \tag{2.23}$$

The vector of 1-forms w is given by:

$$w = [w_1 \; w_2]^T = Q[dx, \; dy, \; d\theta]^T = [-tan\theta dx + dy, \; d\theta]^T \tag{2.24}$$

We have:

$$dw = [dw_1, \; dw_2]^T = \left[-\frac{1}{cos^2\theta}d\theta \wedge dx, \; 0\right]^T \tag{2.25}$$

Which proofs that w is not closed.

We introduce the μ operator:

$$\mu = \begin{bmatrix} 0 & d\left(\frac{x}{cos^2\theta}\right)\wedge \\ 0 & 0 \end{bmatrix} \tag{2.26}$$

Such operator μ verifies $\mu^2 = 0$. Additionally, we have:

$$\mathfrak{d} = \begin{bmatrix} 0 & d\left(\frac{1}{cos^2}dx + 2\frac{xsin\theta}{cos^3\theta}d\theta\right) \wedge \\ 0 & 0 \end{bmatrix} \tag{2.27}$$

By componentwise identification:

$$M = \begin{bmatrix} 1 & -\frac{x}{cos^2\theta} \\ 0 & 1 \end{bmatrix} \tag{2.28}$$

We compute now the 1-form as follows:

$$Mw = \begin{bmatrix} \tan\theta dx + dy - \frac{x}{cos^2\theta}d\theta \\ d\theta \end{bmatrix} \tag{2.29}$$

This 1-form is closed, and the flat output vector is:

$$\begin{aligned} z_1 &= y - xtan\theta \\ z_2 &= \theta \end{aligned} \tag{2.30}$$

This method inspired the proposed FTC approach, in fact the decomposition of $P(F)$ is not unique, an infinity number of them exists, this property could be exploited for FDI, since this will provide redundancy and will increase the number of residual signals, additionally this redundancy could be used to reconfigure the system after fault.

2.5 ANALYTICAL REDUNDANCY BY FLATNESS-BASED APPROACH

Analytical redundancy can be afforded thanks to the main property of the flat systems, which dictates that any input or state can be written as functions of the flat outputs. This provides in a straight manner the redundancy needed to compute the residues, which will indicate the presence of a fault, it will be close to zero if no fault is present and different of it, if a fault affects the system.

Residues are computed by simply comparing the measured variable versus the estimated using the differentially flat equations. This approach is depicted in the figure 2.3, the main advantage of this method is the fact that the estimations of the states and the inputs are only functions of the flat outputs. This phenomenon helps to determine the size of a fault, see for example [49] and [72].

2.5.1 FLATNESS-BASED FAULT DETECTION AND ISOLATION

The main idea of the proposed approach is based in the principle that the set of flat outputs are not unique, in fact, one can find an infinity number of them (Linked with the matrix M in the algorithm), the idea is to find two or more sets of flat outputs which at least one element inside one of the vectors is not an algebraic combination of the first, this action will increase the number of residues, additionally those will be decoupled between them, by consequence this could increase the possibilities of isolate every single fault. Furthermore we consider that the flat outputs are directly states of the system or a linear combination of them and they are consider measured or at least estimated.

Let us consider a nonlinear flat model of dimension n, and m control inputs, with z_α as first set of flat outputs, which corresponds to m components of the state vector, also suppose that the full state is measured, it is always possible to compute n residues:

- n - m state residues, because the full state is supposed to be measured.
- m control inputs residues.

The residual signals are computed by using

$$r^i_{jx} = x_{mk} - \hat{x}_k \tag{2.31}$$

$$r^i_{ju} = u_{ml} - \hat{u}_l \tag{2.32}$$

where x_{mk} and u_{mk} are the k_{th} measured state and control input respectively and $\hat{x}_k$ and $\hat{u}_k$ are the k_{th} state and control input calculated using the differentially flat equations, i is the identifier of the set of flat outputs. In order to clarify the proposed approach, suppose that we have a nonlinear system composed by four states, $[x_1\ x_2\ x_3\ x_4]^T \in n$ and two control inputs $[u_1\ u_2]^T \in m$, as depicted in definition 1.1, the number of control inputs are equal to the number of flat outputs, by consequence $[z] \in m$, suppose too that the nonlinear system is flat and additionnally we can find not one but two set of flat outputs, for instance $z_\alpha = [z_{\alpha 1}\ z_{\alpha 2}]^T = [x_1\ x_2]^T \in m$ and $z_\beta = [z_{\beta 1}\ z_{\beta 2}]^T = [x_3\ x_4]^T \in m$.

In order to show the advantage of founding two sets of flat outputs in which at least one element present in the β set is not an algebraic combination of an element of the α set, the approach is divided in two cases. Fig. 2.5.1 shows the FDI schema.

2.5.2 CASE A: n RESIDUES

Assume now, that only z_α vector exists, this hypothesis implies that:

- The maximal number of residues is four.
- Sensor faults not affecting flat outputs can be isolated depending on the system.
- Flat output sensor faults can be detected but cannot be isolated.

The n residues are obtained as follows:

$$\begin{bmatrix} r^\alpha_{1x} \\ r^\alpha_{2x} \\ r^\alpha_{1u} \\ r^\alpha_{2u} \end{bmatrix} = \begin{bmatrix} x_{m3} \\ x_{m4} \\ u_{m1} \\ u_{m2} \end{bmatrix} - \begin{bmatrix} \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (e_3)^T \\ \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (e_4)^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (c_1)^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (c_2)^T \end{bmatrix} \tag{2.33}$$

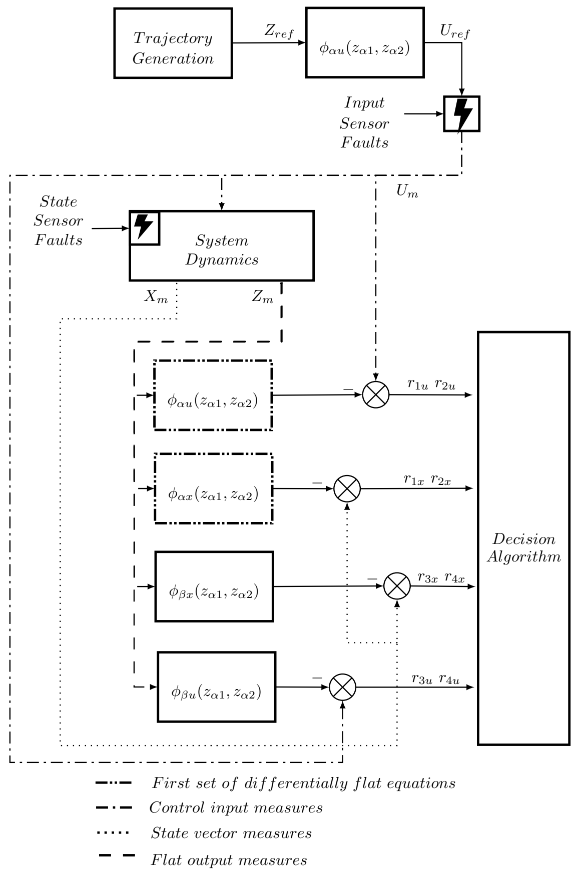

Figure 2.3: Detection diagram

Fault	r^{α}_{1x}	r^{α}_{2x}	r^{α}_{1u}	r^{α}_{2u}
F_{x1}	1	1	1	1
F_{x2}	1	1	1	1
F_{x3}	1	0	0	0
F_{x4}	0	1	0	0
F_{u1}	0	0	1	0
F_{u2}	0	0	0	1

Table 2.2: n residues matrix

Where $e_a \in \mathbb{R}^n$, $e_a = 0$, $\forall a \neq k$,$e_a = 1 \Leftrightarrow a = k$, $a = [1, 2, ..n]$ and $c_b \in \mathbb{R}^m$, $c_b = 0$, $\forall b \neq l$,$c_b = 1 \Leftrightarrow b = l$, $b = [1, 2, ..m]$. Observing in detail equation 2.33, it is straightforward to see that if a fault affects the state measure of (x_{m3}), the residual r^{α}_{1x} will be affected, the rest of residues are independent of this measure, so they will not be affected by the fault. A fault affecting the other state measure or the actuators can be treated in the same manner.

When a fault affects one of the flat outputs, all the residues will be affected, by consequence the fault can be detected but it cannot be isolated. The residues matrix is presented in table 2.2.

2.5.3 CASE B: $n + n$ RESIDUES

Suppose now, that two sets of flat outputs are found, (z_α and z_β), this hypothesis denotes that:

- The maximal number of residues is eight.
- Sensor faults not affecting flat outputs can be detected and isolated.
- Unfaulty versions of the algebraically independent flat outputs could be computed. This property is specially useful to reconfigure the system after fault, this method will be developed in section 2.6.

Fault	r^{α}_{1x}	r^{α}_{2x}	r^{α}_{1u}	r^{α}_{2u}	r^{β}_{1x}	r^{β}_{2x}	r^{β}_{1u}	r^{β}_{2u}
F_{x1}	1	1	1	1	1	0	0	0
F_{x2}	1	1	1	1	0	1	0	0
F_{x3}	1	0	0	0	1	1	1	1
F_{x4}	0	1	0	0	1	1	1	1
F_{u1}	0	0	1	0	0	0	1	0
F_{u2}	0	0	0	1	0	0	0	1

Table 2.3: $n+n$ residues matrix

Using the two algebraically independent set of flat outputs, eight residues are computed, those are the next:

$$\begin{bmatrix} r^{\alpha}_{1x} \\ r^{\alpha}_{2x} \\ r^{\alpha}_{1u} \\ r^{\alpha}_{2u} \\ r^{\beta}_{1x} \\ r^{\beta}_{2x} \\ r^{\beta}_{1u} \\ r^{\beta}_{2u} \end{bmatrix} = \begin{bmatrix} x_{m3} \\ x_{m4} \\ u_{m1} \\ u_{m2} \\ x_{m1} \\ x_{m2} \\ u_{m1} \\ u_{m2} \end{bmatrix} - \begin{bmatrix} \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (e_3)^T \\ \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (e_4)^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (c_1)^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ (c_2)^T \\ \phi_{\beta x}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ (e_1)^T \\ \phi_{\beta x}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ (e_2)^T \\ \phi_{\beta u}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ (c_1)^T \\ \phi_{\beta u}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ (c_2)^T \end{bmatrix} \tag{2.34}$$

Where $e_a \in \mathbb{R}^n$, $e_a = 0$, $\forall a \neq k$,$e_a = 1 \Leftrightarrow a = k$, $a = [1, 2, ..n]$ and $c_b \in \mathbb{R}^m$, $c_b = 0$, $\forall b \neq l$,$c_b = 1 \Leftrightarrow b = l$, $b = [1, 2, ..m]$.

This time if a fault affects x_1, all the z_α residues and r^{β}_{1x} will be triggered; this time the fault is detected and isolated, the same principle can be now applied to every single fault affecting the system, either sensor or actuator faults. Table 2.3 presents the faults signature belonging to each fault.

2.5.4 Detection robustness

For this work the fault detection is achieved by simply comparing the residual amplitude versus a fixed detection threshold.

The amplitude of the detection threshold is fixed by running series of faulty free simulations of the system. The simulations are run with a $\pm$ 10 % individual variation of the parameters of the system, two extra simulations are realized, one varying $+10$ % all the parameters and other varying them -10 %.

Finally the amplitude of the detection threshold is fixed by selecting the worst case among all the results of the simulations, plus a security marge. Such marge is added in order to avoid false alarms caused by the measure noise or modeling errors.

2.5.5 Derivatives estimation

In order to compute the system states and the control inputs of the system, and consequently the residual signals, the time derivatives of the flat outputs of the system has to be estimated.

In this work a high-gain observer [74] is used to evaluate the time derivative of noisy signals.

In order to improve the performance of the high-gain observer, a low-pass filter is synthesized, the filter order is fixed regarding the maximal derivative used in the differentially flat equations, hence a better noise filtering is obtained. Let us define the equation of the high-gain observer:

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u \tag{2.35}$$

Where:

$$\hat{A} = \begin{bmatrix} -\zeta_1/\epsilon & 1 & ... & ... & 0 \\ -\zeta_2/\epsilon^2 & 0 & 1 & ... & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ -\zeta_{n-1}/\epsilon^{n-1} & ... & ... & 0 & 1 \\ -\zeta_n/\epsilon^n & ... & ... & ... & 0 \end{bmatrix} \tag{2.36}$$

And:

$$\hat{B} = \left[\begin{array}{ccccc} -\zeta_1/\epsilon & -\zeta_2/\epsilon^2 & \ldots & -\zeta_{n-1}/\epsilon^{n-1} & -\zeta_n/\epsilon^n \end{array} \right]^T \tag{2.37}$$

The polynomial $S^n + \zeta_1 S^{n-1} + ... + \zeta_{n-1}S + \zeta_n$ is Hurwitz and $\epsilon << 0$. The transfer function from u to $\hat{x}$ when $\epsilon \Rightarrow 0$ is $T(s) = [1\ S\ ...\ S^{n-2}\ S^{n-1}]^T$, the system acts as a differentiator under the consideration that the input u is continuous and derivable. In this case the $n-1$ derivatives are obtained directly from the state vector.

A possible selection of the coefficients $\zeta_i(i = 1, \cdots n)$ is in such a way that the frequency bandwidth of the signal to be derivated is in the frequency bandwidth of the filter $1/(S^n + \zeta_1 S^{n-1} + \cdots + \zeta_{n-1}S + \zeta_n)$ and the ϵ small enough.

2.6 CONTROL RECONFIGURATION FOR DIFFERENTIALLY FLAT SYSTEMS

In order to complement the FTC strategy, control accommodation or control reconfiguration is needed, see 1.9. This work is focused in control reconfiguration,those techniques has as principal characteristic that they keep the nominal controller synthesized during the design phase. This property permits to reduce the response time to a fault. This metodology has as principal characteristic the fact that both stages FDI and reconfiguration are melted in the same block, this characteristic will reduce the time response after fault and could reduce the computational charge of the processor. The goal here is to hide the fault to the controller by changing the faulty reference by an unfaulty one. Fig. 2.4 shows the reconfiguration schema.

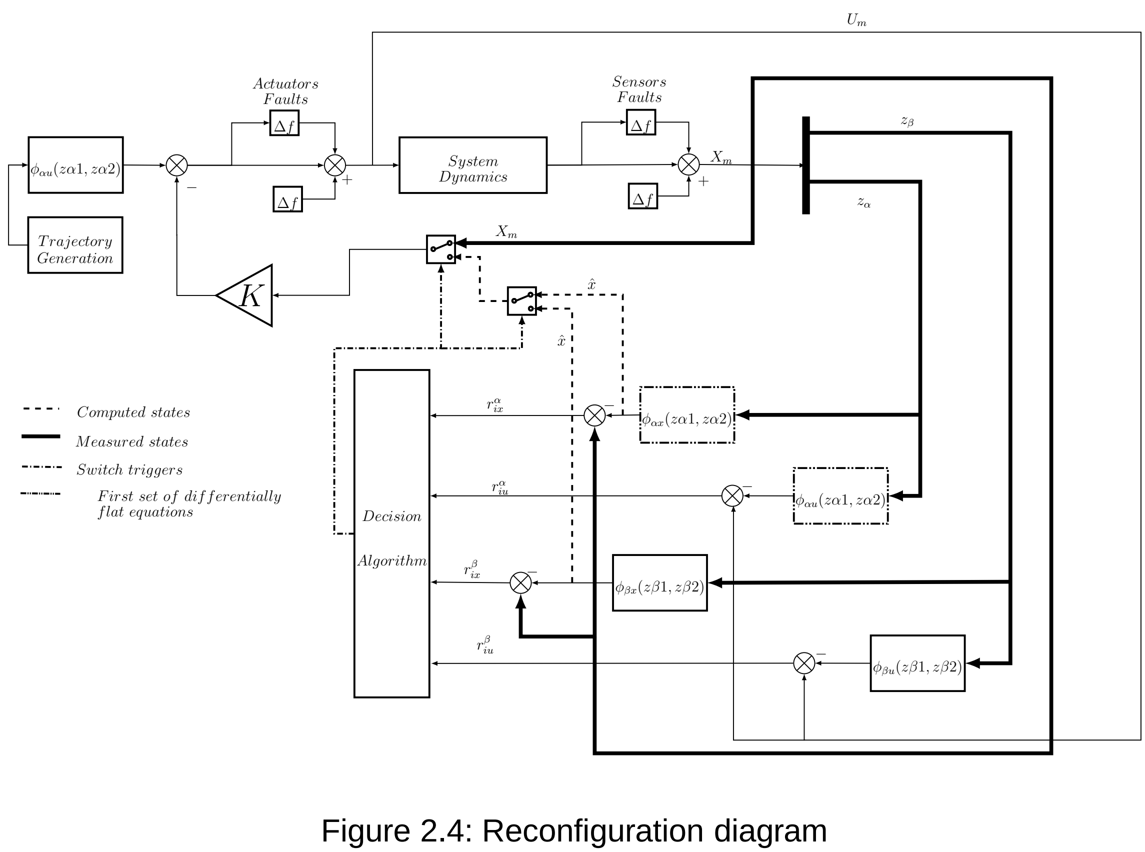

Figure 2.4: Reconfiguration diagram

Let us retake the example **??**, one more time in order to show how founding two or more algebraically independent flat outputs will help to improve the reconfiguration technique, the method will be divided in two different cases.

2.6.1 Case A: Partial reconfiguration

Partial reconfiguration is achieved, if only one set of flat outputs is found, for example $z\alpha$, as explained in section 2.5.2, faults affecting flat outputs cannot be isolated, see table 2.2, however faults affecting measuring sensors can be detected and isolated. Thanks to the properties of the flat systems and the fact that the flat outputs are considered fault-free at any time, we can compute the rest of system states. Those signals can then be used to reconfigure the system.

Empirically the number of reconfigurable faults can be obtained by using the next formula:

$$N_{FLAR} = (FOS)(n - m) \tag{2.38}$$

Where FOS is the number of sets of flat outputs found, n the state dimension and m the number of control inputs. For instance for our example the number of redundant signals is $(1)(4-2) = 2$, which are in fact the two states that are not flat outputs x_3 and x_4.

2.6.2 Case B: Full reconfiguration

Suppose now that two sets of flat outputs are found, $z\alpha$ and $z\beta$. Using the $n + n$ residues every single fault can be detected and isolated, and additionally, the unfaulty set of flat outputs can be used to estimate the faulty state, and then, use this new version to feed the controller and reconfigure the system. See Fig. 2.4.

Let us analyze a fault affecting x_1, if we see in detail the equation 2.34 it is straightforward to see that all the equations containing $z1\alpha$ in the right hand will modify their shape. Since the fault affects the measure of the first state, the first residue r_{1x}^{α} will be affected as well, however a faulty-free version of x_1 is computed using the measures of the z_β set. This signal could be used to hide the fault to the controller. Each fault could be treated in the same manner, which proofs that the system is fully reconfigurable.

2.7 CONCLUSION

This chapter presents the proposed approach for FTC. Additive and multiplicative faults affecting sensors and actuators can be treated in the same manner for detection and isolation. Active reconfiguration is only carried out for sensor faults, actuators faults are rejected by the controller.

The main advantage of the proposed approach is the fact that it melts the FDI process together with the reconfiguration, this action adds simplicity during the design and could reduce the time response to a fault and the computational charge as well.

The next chapter is devoted to investigate the feasibility of the proposed approach, for this, two nonlinear systems will be considered. An unmanned quadrotor and the three tank system, both of them belongs to the group of flat systems, the technique can be applied in a partial manner in the unmanned quadrotor, and will be applied completely in the three tank system.

CHAPTER 3

FAULT TOLERANT CONTROL: APPLICATIONS

Abstract:

In this chapter feasibility of the proposed approach is investigated in two nonlinear systems, first it is applied in a partial way to an unmanned quadrotor and in a second time the full technique is applied to a three tank system.

3.1 INTRODUCTION

This chapter is devoted to investigate the feasibility of the proposed FTC approach. It is divided in two main parts, the first one is devoted to present the nonlinear model of an unmanned quadrotor, for this nonlinear system the FTC technique is not applied entirely, since this system does not meet all the necessary conditions, however as explained in sections 2.5.2 and 2.6.1 the technique can be applied in a partial manner.

The second part is devoted to apply the technique in a three tank system, this system in contrast to the UAV meets all the necessary conditions to exploit at maximum the proposed approach.

3.2 UNMANNED QUADROTOR

The American Institute of Aeronautics and Astronautics (AIAA) defines an unmanned aerial vehicle (UAV) as “an aircraft which is designed or modified, not to carry a human pilot and is operated through electronic input initiated by the flight controller or by an onboard autonomous flight management control system that does not require flight controller intervention” [63]. See Fig. 3.1 The most important characteristic of this kind of vehicles is that they can be recovered at the end of the mission, this property excludes rockets, missiles, shells, etc. The UAV’s have been serving the army since the decade of 90’s, however, thanks to their versatility, and the progress in electronics manufacturing, they are nowadays being used in civil applications, [20] for example:

- Remote sensing and earth science research, [33].
- Search and rescue in human hostile zones (e.g., radiation zones, unstable zones after an earthquake).
- Weather monitoring.

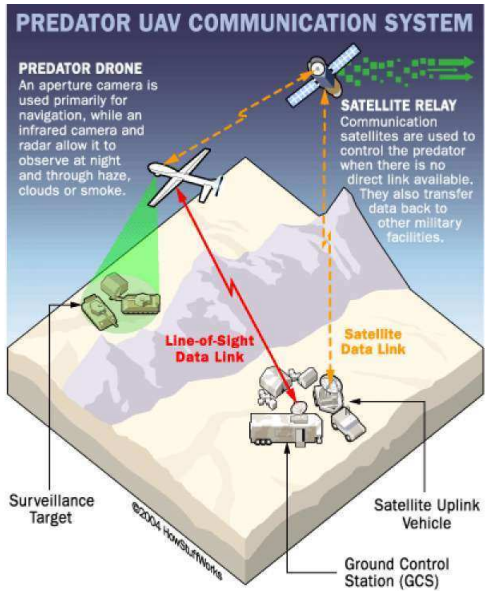

Figure 3.1: UAV communication system

- Crop spraying and dusting.
- Fire fighting, [6].
- Communications networks, [16].

The European Association of Unmanned Vehicles Systems (EUROUVS) has drawn up a clasiffication of uav systems based on such parameters as flight altitude, endurance, speed, maximum take off weight, size, etc., this classification is shown in table 3.1.

	Category (Acronym)	Maximum take-off Weight (kg)	Maximum flight altitude (m)	Endurance (hours)	Data link range (Km)	Example	
						Missions	Systems
Micro/ mini UAV's	Micro (MAV)	0.10	250	1	<10	Scouting, surveillance inside buildings.	Black widow, Microstar, Fancopter, Mosquito.
	Mini	<30	150-300	<2	<10	Film and broadcast ind., agriculture, pollution measurements, communications relay.	Mikado, Aladin, Tracker, Dragon eye, Raven, Skorpio, Robocopter, Pointer II, YH-300SL.
Tactical UAV's	Close Range (CR)	150	3,000	2-4	10-30	Mine detection, search and rescue.	Observer I, Phantom, Copter 4 Robocopter 300, Camcopter.
	Short Range (SR)	200	3,000	3-6	30-70	mine detection.	Luna, SilverFox, EyeView, Hornet.
	Long Range (LR)	-	5,000	6-13	200-500	Communications relay.	Hunter, Vigilante 502.
	Endurance (EN)	500-1500	5,000-8,000	12-24	>500	Battle damage assessment.	Aerosonde, Shadow 600.
	Medium Altitude, Long Endurance (MALE)	1,000-1,500	5,000-8,000	24-48	>500	Weapons delivery, Communications relay.	Skyforce, Heron TP, MQ-1 Predator, Darkstar. Eagle 1 and 2,
Strategic UAV's	High Altitude, Long Endurance (HALE)	2,500-12,500	15,000-20,000	24-48	>2000	boost phase intercept launch vehicle, airport security.	Global Hawk, Raptor, Condor, Theseus, Helios, Libellule, EuroHawk.
Special Task UAV's	Lethal (LET)	250	3,000-4,000	3-4	300	Anti-radar,anti-aircraft.	MALI, Harpy, Lark, Marula.
	Decoys (DEC)	250	50-5,000	<4	0-500	Aerial and naval deception.	Flyrt, MALD, Nulka, ITALD, Chukar.
	Stratospheric (Strato)	-	20,000-30,000	>48	$>2,000$	-	Pegasus.
	Exo-stratospheric (EXO)	-	$>30,000$	-	-	-	MarsFlyer, MAC-1.

Table 3.1: UAV's Classification

3.2.1 NONLINEAR MODEL

The operation of the quadrotor is fairly simple, the position ($\xi = x, y, z$) and the orientation ($\eta = \psi, \theta, \phi$) desired are achieved by independently varying the speed and torque of the four rotors, see figure 3.2.

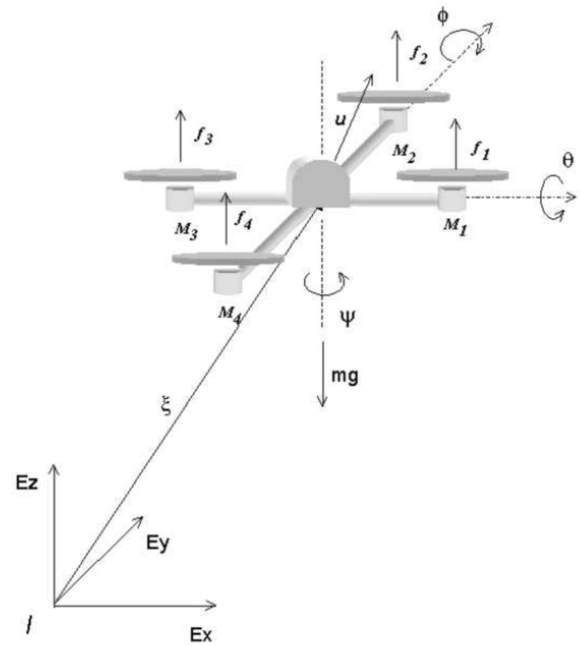

Figure 3.2: Quadrotor schema

The vertical movement is obtained by adding the lift forces generated for each rotor, in order to avoid that the helicopter turns over the z axis, two rotors turn in the clockwise sense (rotors 2 and 4) and the two others turn in the counter-clockwise, this configuration cancel the horizontal moment of the helicopter, which is specially helpful during the hover position. The pitch moment (θ) is achieved by varying the rotation speeds of the rotors 1 and 3, the roll (ϕ) is obtained by varying the rotation speeds of the rotors 2 and 4 and finally the yaw moment (ψ) is obtained from the torque resulting from the substracting of the clockwise (rotors 2 and 4) and the counterclockwise (rotors 1 and 3).

The nonlinear model can be obtained by using the motion equations of Euler-Lagrange. The Lagrangian (L) is defined as the addition of the kinetics (T) and the potential (U) energies.

$$L = T_{translation} + T_{rotation} + U \tag{3.1}$$

Where:

$$T_{translation} = \frac{m}{2}\dot{\xi}^T\dot{\xi} \tag{3.2}$$

$$T_{rotation} = \frac{1}{2}\Omega^T I\Omega = \frac{1}{2}\dot{\eta}^T\mathbb{J}\dot{\eta} \tag{3.3}$$

Where m is the mass of the helicopter and

$$\Omega = \begin{bmatrix} \dot{\phi} - \dot{\psi}sin\theta \\ \dot{\theta}cos\phi + \dot{\psi}cos\theta sin\phi \\ \dot{\psi}cos\theta cos\phi - \dot{\theta}sin\phi \end{bmatrix} \tag{3.4}$$

$$\mathbb{J} = \mathbb{J}(\eta) = W_\eta^T I W_\eta \tag{3.5}$$

$$W_\eta = \begin{bmatrix} -sin\theta & 0 & 1 \\ cos\theta sin\theta & cos\theta & 0 \\ cos\theta cos\phi & -sin\phi & 0 \end{bmatrix} \tag{3.6}$$

The inertial matrix has elements only in the principal diagonal, because the aircraft is considered symmetric.

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{3.7}$$

The potential energy is equal to:

$$U = -mgz \tag{3.8}$$

Assembling all the parts the equation (3.1) can be wrote in the next manner:

$$L = \frac{m}{2}\dot{\xi}^T\dot{\xi} + \frac{1}{2}\dot{\eta}^T\mathbb{J}\dot{\eta} - mgz \tag{3.9}$$

Which satisfies the Euler-Lagrange equation.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \left(\frac{\partial L}{\partial q}\right) = \mathbf{F}_L \tag{3.10}$$

Where $\mathbf{F}_L$ stands for the forces and moments applied to the body frame of the aircraft. Due that the Lagrangian does not contain cross terms combining the position and the orientation, the Euler-Lagrange equation can be divided in the dynamics of the ξ and η coordinates individually.

By this manner the dynamic model can be wrote as follows:

$$\begin{bmatrix} \frac{d}{dt}\left(\frac{\partial Ltranslation}{\partial \dot{\xi}}\right) - \frac{\partial Ltranslation}{\partial \xi} & 0 \\ 0 & \frac{d}{dt}\left(\frac{\partial Lrotation}{\partial \dot{\eta}}\right) - \frac{\partial Lrotation}{\partial \eta} \end{bmatrix} = \begin{bmatrix} f \\ \tau \end{bmatrix} \tag{3.11}$$

Where $f = R\mathbf{F}_L$, is the force applied to the aircraft due to the lift generated by the four rotors and τ represents the moments of pitch, roll and yaw.

$$\mathbf{F}_L = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix} \tag{3.12}$$

$$u = f_1 + f_2 + f_3 + f_4 \tag{3.13}$$

Where $f_i, i = 1, 2, 3, 4$ is the force produced for each one of the rotor, $f_i = k_i w_i^2$. And:

$$\tau = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} = \begin{bmatrix} \Sigma_{i=1}^4 \tau M_i \\ (f_2 - f_4)l \\ (f_3 - f_1)l \end{bmatrix} \tag{3.14}$$

Where l is the distance between rotors and the center of gravity, and τM_i is the moment produced by the motor i.

Finally, the nonlinear model can be obtained by solving the Euler-Lagrange equations for position and orientation. For position:

$$\frac{d}{dt}\left(\frac{\partial Ltraslacional}{\partial \dot{\xi}}\right) - \frac{\partial Ltraslacional}{\partial \xi} = f \tag{3.15}$$

Substituying the value of $L_{Translational}$ and adding the potential energy because it cause a movement in the z axis, we obtain:

$$\frac{d}{dt}\left(\frac{\partial \frac{m}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + mgz}{\partial \dot{\xi}}\right) - \frac{\partial \frac{m}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + mgz}{\partial \xi} = f \tag{3.16}$$

By computing the derivative we obtain:

$$\frac{d}{dt}\left(\frac{m}{2}(2\dot{x} + 2\dot{y} + 2\dot{z})\right) + 0 - 0 - 0 - mg = f \tag{3.17}$$

Finally computing the time derivative and rearranging in vector form, we obtain the equations related to the position coordinates.

$$f = \begin{bmatrix} m\ddot{x} \\ m\ddot{y} \\ m\ddot{z} - mg \end{bmatrix} \tag{3.18}$$

For the orientation coordinates:

$$\frac{d}{dt}\left(\frac{\partial Lrotacional}{\partial \dot{\eta}}\right) - \frac{\partial Lrotacional}{\partial \eta} \tag{3.19}$$

Substituying we obtain:

$$\frac{d}{dt}\left(\frac{\partial(\frac{1}{2}\dot{\eta}^T \mathbb{J}\dot{\eta})}{\partial \dot{\eta}}\right) - \frac{\partial(\frac{1}{2}\dot{\eta}^T \mathbb{J}\dot{\eta})}{\partial \eta} \tag{3.20}$$

Computing the derivatives:

$$\frac{d}{dt}\left(\frac{1}{2}\left(\frac{\partial \dot{\eta}^T}{\partial \dot{\eta}} + 0 + \dot{\eta}^T \mathbb{J}\frac{\partial \dot{\eta}}{\partial \dot{\eta}}\right)\right) - \frac{1}{2}\left(0 + \frac{\partial}{\partial \eta}(\dot{\eta}^T \mathbb{J}\dot{\eta} + 0\right) = \tau \tag{3.21}$$

Computing the time derivative:

$$\mathbb{J}\ddot{\eta} + \dot{\mathbb{J}}\dot{\eta} - \frac{1}{2}\left(\frac{\partial}{\partial \eta}\left(\dot{\eta}^T \mathbb{J}\dot{\eta}\right)\right) = \tau \tag{3.22}$$

In order to write the equation above in the general form $M(\eta)\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau$ we factorize $\dot{\eta}$ to the right as follows:

$$\mathbb{J}\ddot{\eta} + \left(\dot{\mathbb{J}} - \frac{1}{2}\frac{\partial}{\partial\eta}(\dot{\eta}^T\mathbb{J})\right)\dot{\eta} = \tau \tag{3.23}$$

By this way we can define the coriolis matrix ($C(\eta,\dot{\eta})$) and the inertial matrix in the next manner:

$$C(\eta,\dot{\eta}) = \dot{\mathbb{J}}\dot{\eta} - \frac{1}{2}\left(\frac{\partial}{\partial\eta}\left(\dot{\eta}^T\mathbb{J}\dot{\eta}\right)\right) \tag{3.24}$$

$$M(\eta) = \mathbb{J}(\eta) = W_\eta^T \mathbf{I} W_\eta \tag{3.25}$$

Finally the nonlinear dynamical model of the quadrotor is:

$$f = \begin{bmatrix} m\ddot{x} \\ m\ddot{y} \\ m\ddot{z} - mg \end{bmatrix} \tag{3.26}$$

$$\tau = M(\eta)\ddot{\eta} + C(\eta,\dot{\eta})\dot{\eta} \tag{3.27}$$

In order to simplify the model let us introduce the change of input variables proposed in [7].

$$\tilde{\tau} = \begin{bmatrix} \tilde{\tau_\psi} \\ \tilde{\tau_\theta} \\ \tilde{\tau_\phi} \end{bmatrix} = M(\eta)^{-1}\left(\tau - C(\eta,\dot{\eta})\dot{\eta}\right) \tag{3.28}$$

Where $\tilde{\tau} = \ddot{\eta}$, are the new inputs, after this transformation the nonlinear model becomes:

$$
\begin{aligned}
m\ddot{x} &= -u_1 sin\theta & \ddot{\psi} &= u_2 \\
m\ddot{y} &= u_1 cos\theta sin\phi & \ddot{\theta} &= u_3 \\
m\ddot{z} &= u_1 cos\theta cos\phi - mg & \ddot{\phi} &= u_4
\end{aligned} \tag{3.29}
$$

By this way the nonlinear model is compound by twelve states, $X = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \psi\ \theta\ \phi\ \dot{\psi}\ \dot{\theta}\ \dot{\phi}]^T = [x_1\ x_2 x_3\ x_4\ x_5\ x_6\ x_7\ x_8\ x_9\ x_{10}\ x_{11}\ x_{12}]^T$ and the control inputs are $U = [u_1\ \ddot{\psi}\ \ddot{\theta}\ \ddot{\phi}]^T = [u_1\ u_2\ u_3\ u_4]^T$.

3.2.2 Flat model

The goal in the flatness approach is explicitly express all the states and all the control inputs as functions of the flat outputs and a finite number of its time derivatives, so, from the equations 3.29, and defining the flat outputs as $z\alpha = [x\, y\, z\, \psi]^T$ [14], because we have four control inputs. Besides we can write all the system states as function of the flat outputs $z\alpha$ and its time derivatives as follows:

$$
\begin{aligned}
x &= z_1 & \dot{x} &= \dot{z}_1 \\
y &= z_2 & \dot{y} &= \dot{z}_2 \\
z &= z_3 & \dot{z} &= \dot{z}_3 \\
\psi &= z_4 & \dot{\psi} &= \dot{z}_4 \\
\theta &= asin\left(\frac{m\ddot{z}_1}{-u_1}\right) & \dot{\theta} &= -m\left(\frac{z_1^{(3)}u_1 - u_1^2\ddot{z}_1}{u_1^2\sqrt{\alpha}}\right) \\
\phi &= atan\left(\frac{\ddot{z}_2}{\ddot{z}_3 + g}\right) & \dot{\phi} &= \frac{z_2^{(3)}(\ddot{z}_3 + g) - z_3^{(3)}\ddot{z}_2}{(\ddot{z}_3 + g)^2 + \ddot{z}_2{}^2}
\end{aligned} \tag{3.30}
$$

In a similar way the control inputs are expressed as function of the flat outputs and its time derivatives.

$$
\begin{aligned}
u_1 &= m\sqrt{\ddot{z}_1^2 + \ddot{z}_2^2 + \ddot{z}_3^2 + 2\ddot{z}_3 g + g^2} \\
u_2 &= \ddot{z}_4 \\
u_3 &= -m\left(\frac{\left((z_1^{(4)}u_1 - \ddot{u}_1\ddot{z}_1)u_1^2\sqrt{A}\right) - (z_1^{(3)}u_1 - \ddot{u}_1\ddot{z}_1)\,(C) + 2u_1\dot{u}_1\sqrt{A}}{u_1^4 A}\right)
\end{aligned} \tag{3.31}
$$

$$u_4 = \frac{(z_2^{(4)}(\ddot{z}_3+g)+z_2^{(3)}z_3^{(3)} - z_3^{(4)}\ddot{z}_2 - z_3^{(3)}z_2^{(3)})(z_2^{(3)}(\ddot{z}_3+g) - z^{(3)}\ddot{z}_2) - B}{(\ddot{z}_3+g)^4+(\ddot{z}_2)^4}$$

where

$$A = 1-\left(\frac{m\ddot{z}_1}{u_1}\right)^2 \tag{3.32}$$

$$B = (2(\ddot{z}_3+g)z_3^{(3)}+2\ddot{z}_2z_2^{(3)})(z_2^{(3)}(\ddot{z}_3+g)-z_3^{(3)}\ddot{z}_2) \tag{3.33}$$

$$C = (\frac{-2m\ddot{z}_1(mz_1^{(3)}u_1-\dot{u}_1m\ddot{z}_1)}{u_1\sqrt{\alpha}} \tag{3.34}$$

3.2.3 FLATNESS-BASED FAULT TOLERANT CONTROL OF A QUADROTOR UAV

Additive faults affecting sensors and control inputs (combination of actuators, see (3.14)) are considered. For sensors measuring x_{m5} (θ) and x_{m6} (ϕ) different faults amplitudes are consider, see Table 3.2 such amplitude defines the FTC strategy used to counteract the fault. For sensors measuring flat outputs z_1, z_2 and z_3 an additive fault of one meter is considered. For the flat output z_4, $1°$ extra is applied. Only single faults are considered. Reconfiguration after fault is taken into account only for measuring sensors. Once the fault appears (50 s) it is recurrent until the end of the simulation. Since the FTC strategy needs to know the amplitude of the fault, in order to decide which strategy will be used. For simplicity sake on this work the fault amplitude is supposed perfectly known, by consequence the strategy choice is straightforward. Section 2.5.4 describes how

Fault	Amplitude	Strategy	Amplitude	Strategy	Amplitude	Strategy
F_{x5}	$< 1.8°$	P^1	$> 1.8° < 4.6°$	Rf^1	$>4.6°$	$Re^{1,2}$
F_{x6}	$< 1.8°$	P	$> 1.8° < 3.9°$	Rf	$>3.9°$	Re

Table 3.2: Additives faults for the UAV

[1]P =Passive, Rf =Reconfiguration, Re =Restructuring. [2]This approach is out of the bounds of this work.

the detection threshold is fixed. The parameter that change is the mass (m) of the helicopter, the nominal value is 0.52Kg. The controller in charge of close the loop is an LQR, the matrices Q and R are chosen in order to respect the power bounds of the actuators. The nominal trajectories are created using order 5^{th} polynomials. White noise is added to the signal in order to simulate real operation. High gain observers are used to compute the time derivatives, low-pass filters are coupled with the observers. A trade-off between the time delay caused by the filter and the cut-off frequency needs to be studied in detail. A cut-off frequency very high will not reduce properly the amplitude of the noise, on the other hand, higher the frequency of the filter the time delay induced will be more important, this delay could prevent the use of the reconfiguration, because if the estimated signal is not in phase with the measure the fact that change between references could drive the system to instability.

3.2.3.1 FAULT DETECTION AND ISOLATION

For this particular system only one set of flat outputs is found, by consequence $n = 12$ residues are found, which is in fact the number of states, however, for simplicity sake the time derivatives of the three position states and the three orientation states, x_7 to x_{12} are consider unfaulty at any time, such supposition produces only six residues, which are presented in equation (3.35).

$$\begin{bmatrix} r_{1x} \\ r_{2x} \\ r_{1u} \\ r_{2u} \\ r_{3u} \\ r_{4u} \end{bmatrix} = \begin{bmatrix} x_{m5} \\ u_{m6} \\ u_{m1} \\ u_{m2} \\ u_{m3} \\ u_{m4} \end{bmatrix} - \begin{bmatrix} \phi_x(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [0_{(4)}\, 1\, 0_{(7)}]^T \\ \phi_x(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [0_{(5)}\, 1\, 0_{(6)}]^T \\ \phi_u(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [1\,0\,0\,0]^T \\ \phi_u(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [0\,1\,0\,0]^T \\ \phi_u(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [0\,0\,1\,0]^T \\ \phi_u(z_1, \dot{z}_1, z_2, \dot{z}_2)\ [0\,0\,0\,1]^T \end{bmatrix} \tag{3.35}$$

For faults affecting measuring sensors two different frameworks are considered. See Table 3.2. However for FDI the fault amplitude is not a key parameter, since the fault signature is the same regardless of the amplitude fault. All residues are normalized between -1 and +1, those edges represents the minimal and maximal amplitude of the fault-free threshold.

Let us analyze each individual fault. For instance a fault affecting measure of x displacement should affect each one of the six residues because it is a flat output, however the system is naturally decoupled. By consequence only the residues depending on the x_{m1} measure are affected. Those are r_{1x}, r_{1u} and r_{3u}, see equations (3.30) and (3.31). See Fig. 3.3. This behavior is due to the operation of the UAV, in fact since the axis of the four rotors are fixed to the main frame (cannot tilt) the horizontal displacement can only be obtained by tilting the entire frame in order to move the airplane. By consequence the residues which depends directly of θ are impacted. The residual r_{1u} is affected because it depends of the time derivative of x.

Figure 3.4 shows the residues obtained after a fault on sensor y. This time the fault affects the residues related to ϕ (r_{2x} and r_{4u}). This is due to the same phenomenon presented in the x axis. Once again the residue r_{1u} is affected because it depends on the faulty measure.

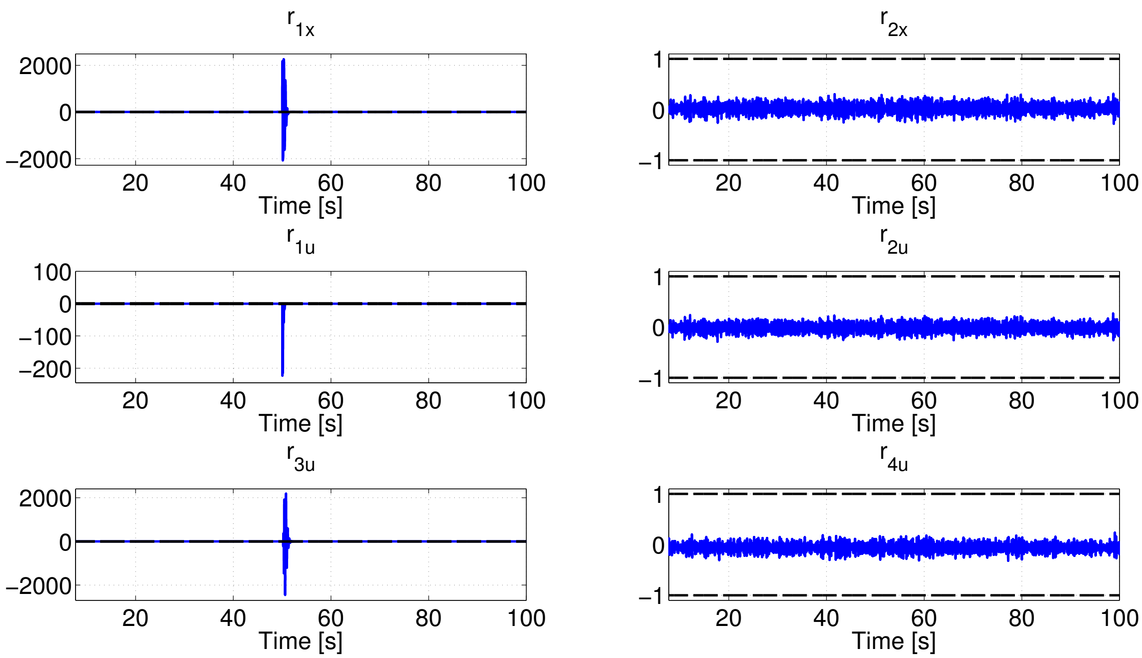

Figure 3.3: Additive fault measure x_1 residues normalized

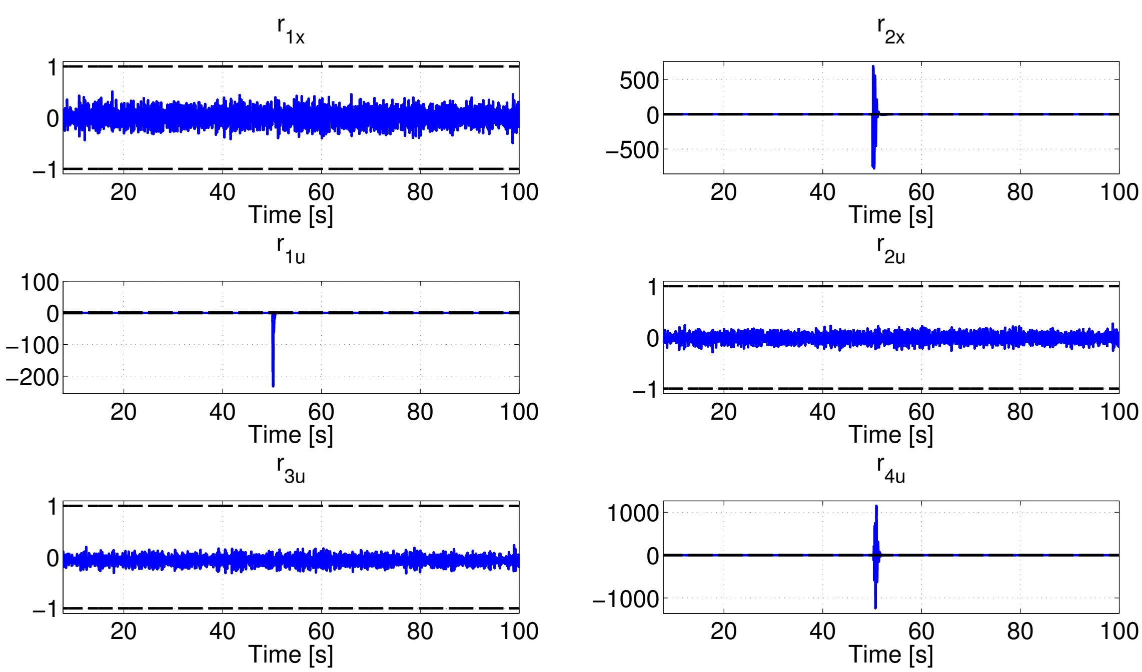

Figure 3.4: Additive fault measure x_2 residues normalized

A fault affecting the high measure (z) will impact five of six residues. Because it is present directly or indirectly in the equations used to estimate the states and the control inputs. See equations (3.30) and (3.31). The residue not affected depends only on the yaw (ψ) movement of the airplane, by consequence residue r_{4u} is not affected by the fault effect.

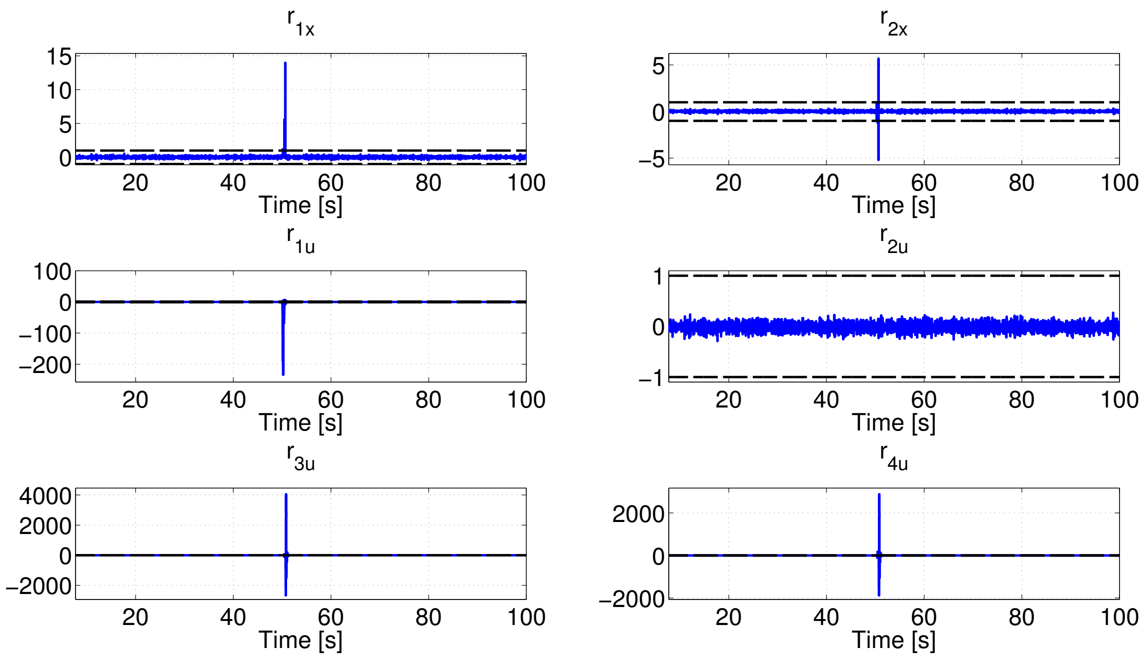

Figure 3.5: Additive fault measure x_3 residues normalized

Finally Fig. 3.6 present the residues for a fault affecting the sensor measuring the yaw angle. Residual r_{4u} is directly related to this measure, by consequence it is triggered.

Fault affecting the pitch angle, x_{m5} will trigger all the residues depending on θ, such residues are r_{1x} and r_{2u}. However, even if the residue r_{1u} is affected indirectly (via the x displacement) the amplitude change is not enough to exceeds the threshold, see Fig. 3.7.

For roll angle x_{m6} is quite similar, it differs in the fact that this time the residue r_{1u} is affected by the y displacement. Fig. 3.8.

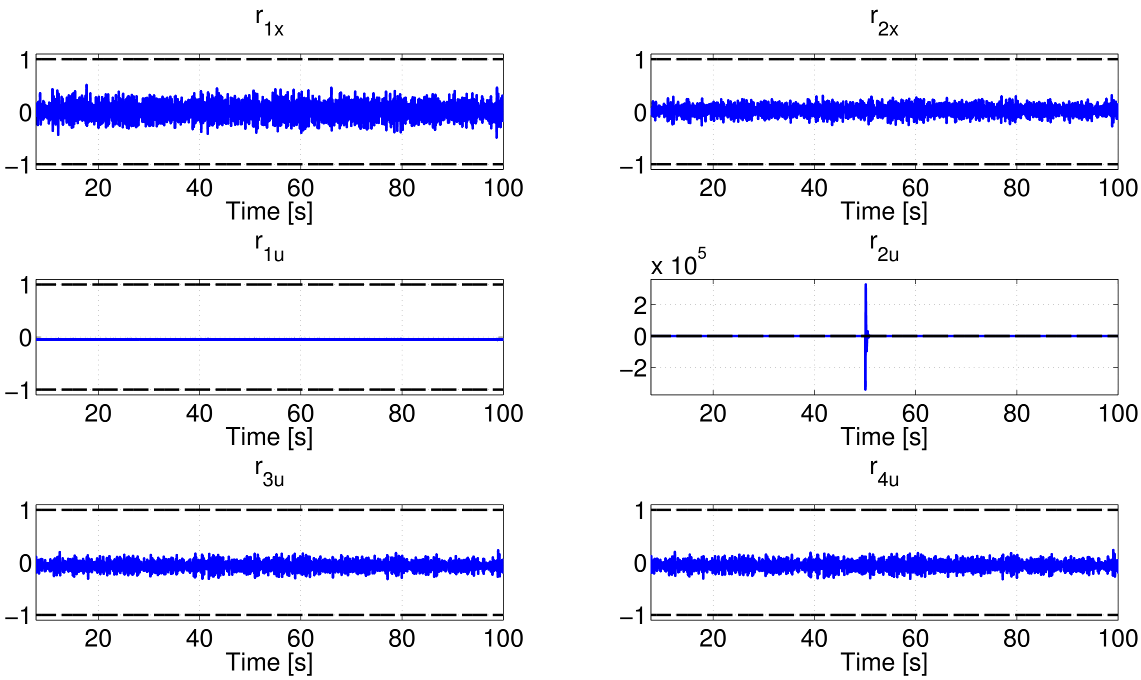

Figure 3.6: Additive fault measure x_4 residues normalized

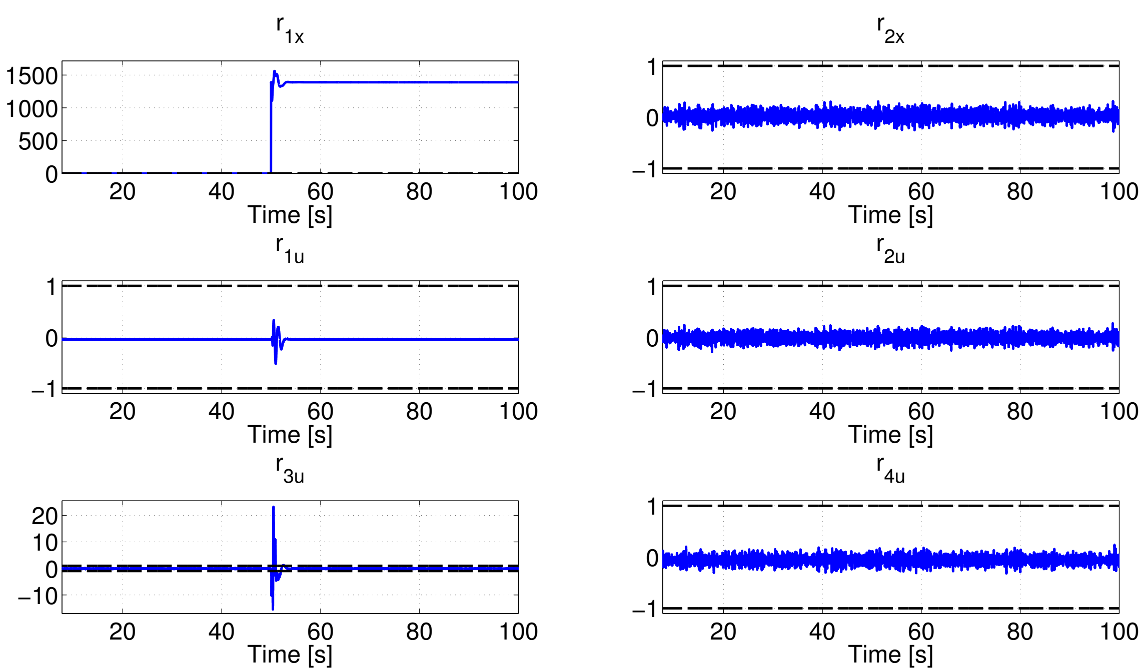

Figure 3.7: Additive fault measure x_5 residues normalized

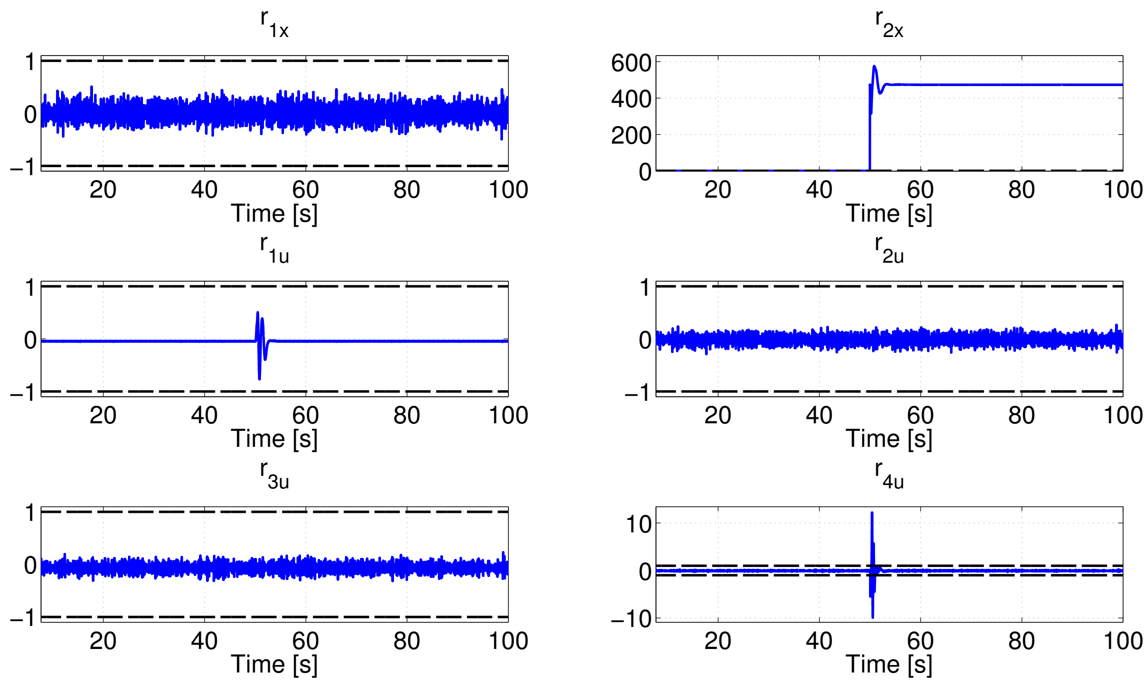

Figure 3.8: Additive fault measure x_6 residues normalized

For control inputs faults, the fault amplitude is equals to a 20 % of the maximal value of the nominal trajectory, see Table 3.2. Faults affecting control input u_1 could be detected and isolated, Fig. 3.9. However faults in the next three control inputs are hidden by the noise. Such faults becomes detectable and isolable if the amplitude is augmented, however even if the movement of the aircraft is completely oversized (displacements of more than 100 meters) the control inputs has as maximum an amplitude of 0.02, by consequence an enormous fault, for instance equal to one, is completely unrealistic. Such faults are not considered.

3.2.3.2 CONTROL RECONFIGURATION

For this section only faults of sensors x_{m5} and x_{m6} are considered. the number of redundant available signals is (1)*(12-4)=8, see 2.38. This number is reduced to two, because as in the FDI part, the states x_7 to x_{12} are considered fault-free at any time, so reconfiguration is not needed.

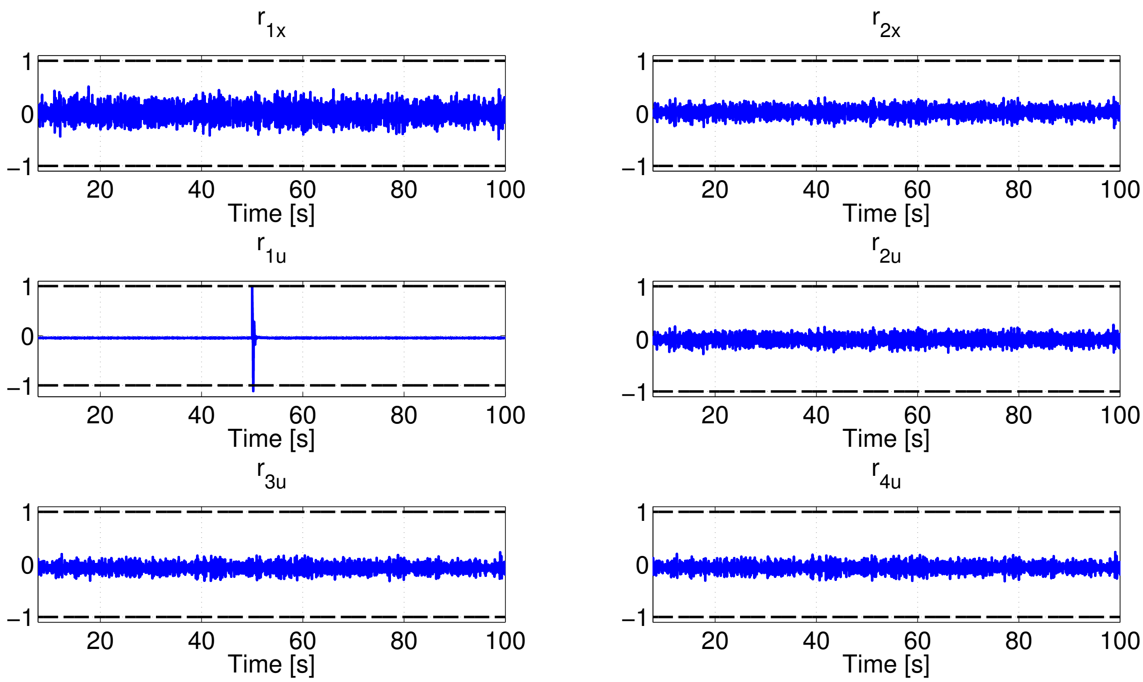

Figure 3.9: Additive fault control input u_1 residues normalized

The goal of the reconfiguration method is to hide the fault to the controller, this is achieved by computing a fault-free reference using the differentially flat equations 3.30. The strategy to change the controller reference is only switch between the signal coming from the sensor and the signal computed with the equation 3.30. Possible instabilities due to the switch effect are not consider in this work.

Figures 3.10 and 3.11 shows the comparison between FTC Passive approach (—), FTC active proposed approach (−· −) and nominal behavior (− − −) for faults affecting θ and ϕ measurement sensors. In the passive case the switch is not activated, the signal coming from sensor stills the same, the fault is rejected by the controller. On the other hand if the amplitude fault exceeds the limits of the passive approach, the switch is triggered in order to change the signal coming from the measuring sensor by the estimation computed with the differentially flat system equations. This action has as consequence the reconfiguration of the control loop. See Figs. 3.10 and 3.11.

Fault	r_{1x}	r_{2x}	r_{1u}	r_{2u}	r_{3u}	r_{4u}
F_{x1}	1	0	1	0	1	0
F_{x2}	0	1	1	0	0	1
F_{x3}	1	1	1	0	1	1
F_{x4}	0	0	0	1	0	0
F_{x5}	1	0	0[3]	0	1	0
F_{x6}	0	1	0[3]	0	0	1
F_{u1}	0	0	1	0	0	0

Table 3.3: Residues matrix Quadrotor UAV

[3]This residue is affected but the amplitude is not enough to exceed the threshold.

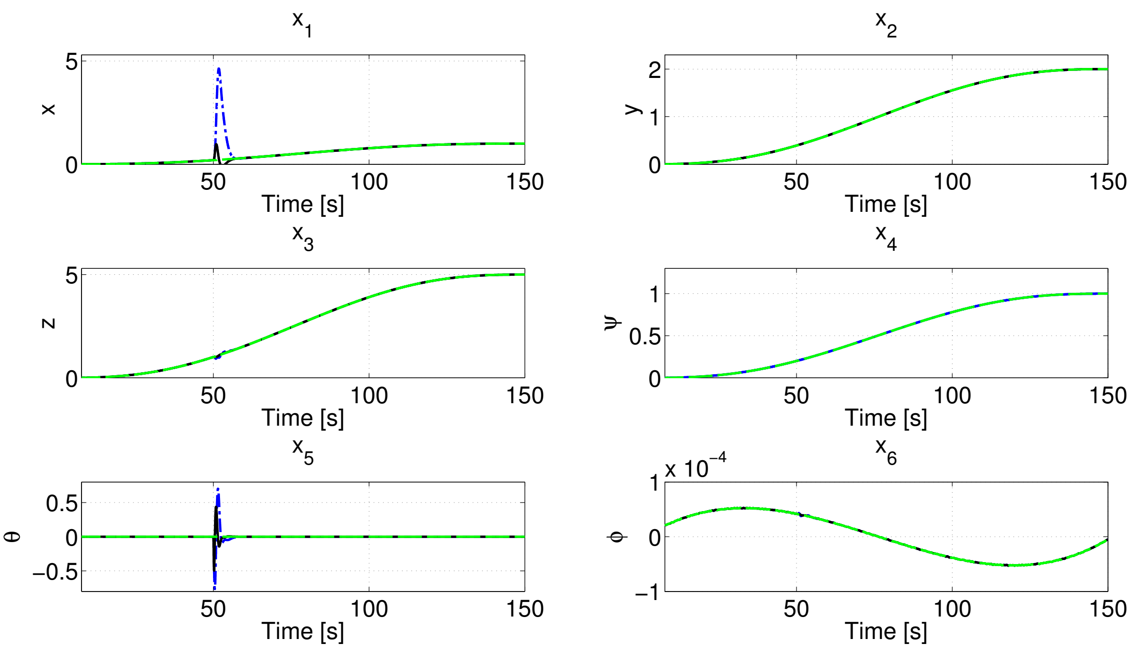

Figure 3.10: Reconfiguration after fault x_5. Passive (—). Proposed approach (—· —). Nominal (— — —).

The effectiveness of the proposed approach presented in the previous figures could be compared versus the Figs. 3.12 and 3.13. It is straightforward to see that if the control is not reconfigured the system became quickly unstable. In both figures the yaw (ψ) is not touched, this phenomenon is explained because the physical decoupling between this angle and the pitch and roll angles.

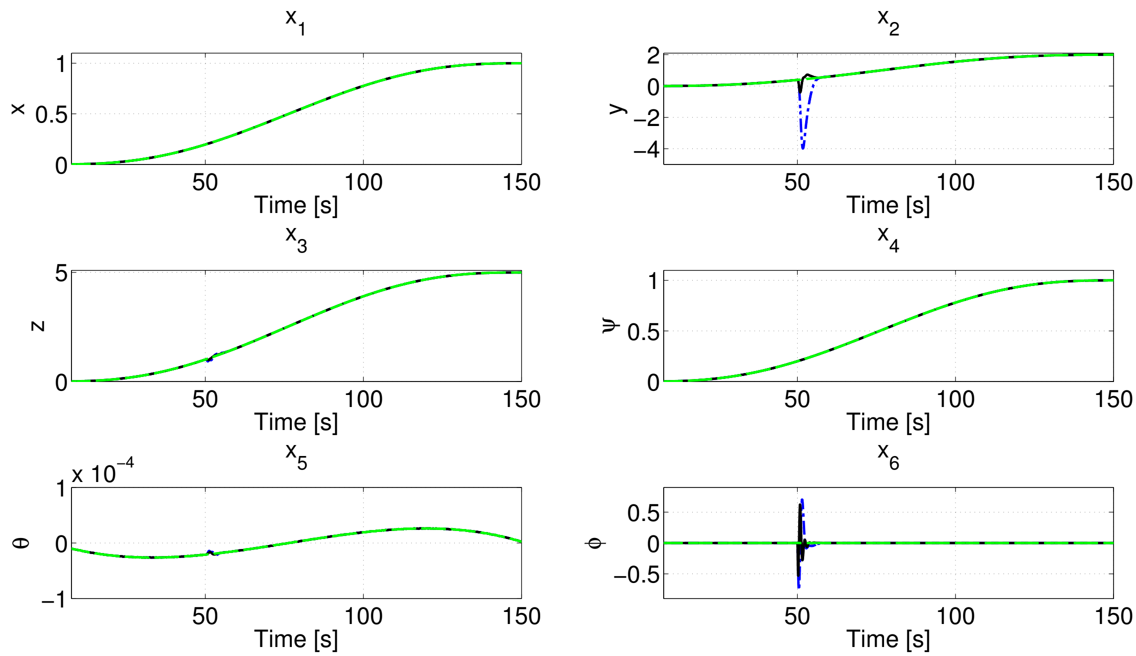

Figure 3.11: Reconfiguration after fault x_6. Passive (—). Proposed approach (–· –). Nominal (– – –).

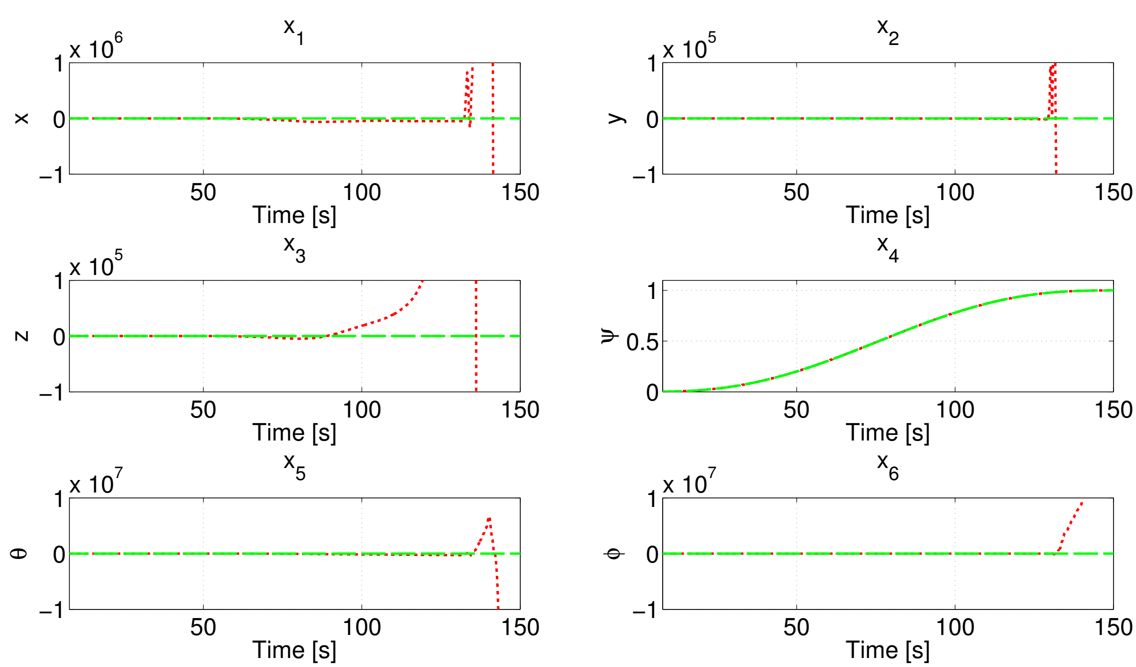

Figure 3.12: Fault affecting x_5. No-reconfiguration (· · ·). Nominal (– – –).

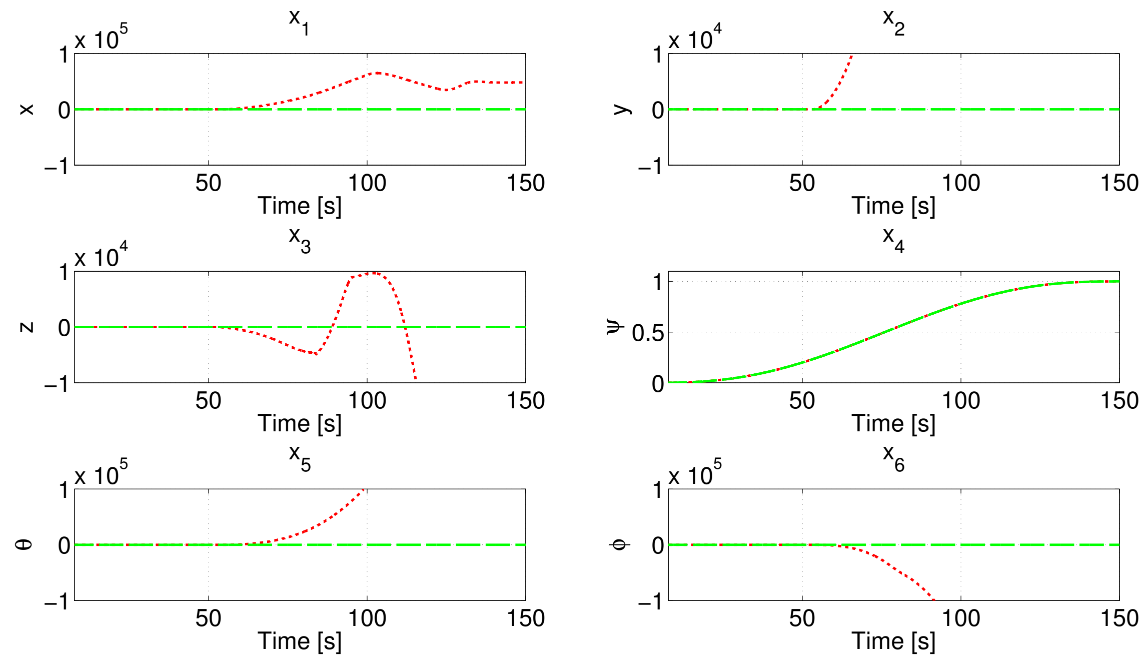

Figure 3.13: Fault x_6. No-reconfiguration ($\cdots$). Nominal ($---$).

3.3 THREE TANK SYSTEM

The system is compose by three tanks connected one next each other, the three of them has the same surface section S, a central reservoir and two inflow pumps. Each tank is linked to the central reservoir by means of a pipe, in which the flow is adjustable manually. The tanks are related with pipes of section S_n. See Fig. 3.14.

3.3.1 NONLINEAR MODEL

The water level inside each tank is proportional to the integral of the flows inside the pipes, by consequence we can write the next equations:

$$\begin{aligned} S\dot{x}_1 &= -Q_{10}(x_1) - Q_{13}(x_1, x_3) + u_1 \\ S\dot{x}_2 &= -Q_{20}(x_2) + Q_{32}(x_2, x_3) + u_2 \\ S\dot{x}_3 &= Q_{13}(x_1, x_3) - Q_{32}(x_2, x_3) - Q_{30}(x_3) \end{aligned} \tag{3.36}$$

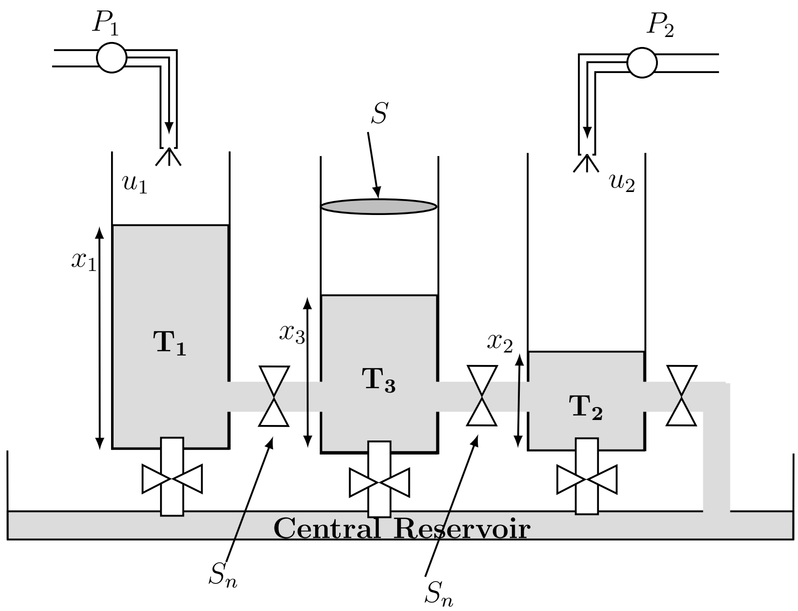

Figure 3.14: Three tank system

Where S is the transverse section of the tanks, x_i, $i = 1,2,3$ water level of each tank, Q_{i0}, $i = 1,2,3$ the outflow between each tank and the central reservoir, Q_{13} and Q_{32} are the outflow between tank 1 and tank 3 and the outflow between tanks 3 and 2 respectively, u_1 and u_2 are the incoming flows of each pump.

The valves connecting tanks one and three with the central reservoir are considered closed, so Q_{10} and Q_{30} are always equals to zero. The flows Q_{13}, Q_{32} and Q_{20} can be expressed as follows:

$$\begin{aligned} Q_{13}(x_1, x_3) &= a_{z1} S_n \sqrt{2g(x_1 - x_3)} \\ Q_{20}(x_2) &= a_{z2} S_n \sqrt{2g(x_2)} \\ Q_{32}(x_2, x_3) &= a_{z3} S_n \sqrt{2g(x_3 - x_2)} \end{aligned} \tag{3.37}$$

Where S_n represents the transverse section of the pipes connecting the tanks and a_{zr}, $r = 1,2,3$ represents the flow coefficients.

3.3.2 FLAT MODEL

The flat model is computed by defining x_1 and x_3 as flat outputs, $z_\alpha = [x_1\ x_3]^T$, so the differentially flat equations can be writen as follows:

$$
\begin{aligned}
x_1^\alpha &= z_{\alpha 1} \\
x_2^\alpha &= z_{\alpha 2} - \frac{1}{2g}\left(\frac{a_{z1}S_n\sqrt{2g(z_{\alpha 1}-z_{\alpha 2})}-S\dot{z}_{\alpha 2}}{a_{z3}S_n}\right)^2 \\
x_3^\alpha &= z_{\alpha 2} \\
u_1^\alpha &= S\dot{z}_{\alpha 1} + a_{z1}S_n\sqrt{2g(z_{\alpha 1}-z_{\alpha 2})} \\
u_2^\alpha &= S\dot{x}_2^\alpha - a_{z3}S_n\sqrt{2g(z_{\alpha 2}-x_2^\alpha)} + a_{z2}S_n\sqrt{2gx_2^\alpha}
\end{aligned}
\tag{3.38}
$$

$$\phi_{\alpha x}(z_{\alpha 1}, z_{\alpha 2}) = \left[\ x_1^\alpha\ x_2^\alpha\ x_3^\alpha\ \right]^T \tag{3.39}$$

$$\phi_{\alpha u}(z_{\alpha 1}, z_{\alpha 2}) = \left[\ u_1^\alpha\ u_2^\alpha\ \right]^T \tag{3.40}$$

As mentioned above the flat vector for this system, is not unique, so, it is possible to use $z_\beta = [x_2\ x_3]^T$ in order to compute another set of differentially flat equations.

$$
\begin{aligned}
x_1^\beta &= z_{\beta 2} + \frac{1}{2g}\left(\frac{a_{z3}S_n\sqrt{2g(z_{\beta 2}-z_{\beta 1})}+S\dot{z}_{\beta 2}}{a_{z1}S_n}\right)^2 \\
x_2^\beta &= z_{\beta 1} \\
x_3^\beta &= z_{\beta 2} \\
u_1^\beta &= S\dot{x}_1^\beta + a_{z1}S_n\sqrt{2g(x_1^\beta - z_{\beta 2})} \\
u_2^\beta &= S\dot{z}_{\beta 1} - a_{z3}S_n\sqrt{2g(z_{\beta 2}-z_{\beta 1})} + a_{z2}S_n\sqrt{2gz_{\beta 1}}
\end{aligned}
\tag{3.41}
$$

$$\phi_{\beta x}(z_{\beta 1}, z_{\beta 2}) = \left[\ x_1^\beta\ x_2^\beta\ x_3^\beta\ \right]^T \tag{3.42}$$

$$\phi_{\beta u}(z_{\beta 1}, z_{\beta 2}) = \left[\ u_1^\beta\ u_2^\beta\ \right]^T \tag{3.43}$$

3.3.3 FLATNESS-BASED FAULT TOLERANT CONTROL OF A THREE TANK SYSTEM

This time additive and multiplicative faults are considered, such faults can affect sensors and actuators. Faults affecting the actuators are consider rejected by the controller, by consequence reconfiguration is not needed.

For additive faults, a $+8cm$ fault is considered for sensors and for flow actuators an extra flow of $0.8 * 10^{-5} m^3/s$ is added. Concerning multiplicative faults a $20\,\%$ failure is considered for sensors and actuators. Only one single fault is considered at any time, once the fault appears (at 250 s) it is recurrent until the end of the simulation.

The detection threshold is fixed as explained in section 2.5.4, if it is exceeded the fault is consider detected. The parameters varying for this system are the flow coefficients, a_{z1} and a_{z3}. Nominal values are equal to 0.75 and 0.76 respectively, the transverse section of the tanks S and the transverse section of the connecting pipes S_n are $15.4 * 10^{-3}$ and $5 * 10^{-5}$ respectively. Both sections remains with out changes during the process to fix the threshold and the simulations.

The control loop is closed with an state feedback LQR controller, the matrix Q and R are chosen in order to respect the mechanical limits of the pumps and avoid outflow peaks. Additionally, saturation functions are connected to both pumps, an integral action on level measures in tanks 1 and 2 is added in order to eliminate the steady state error.

The nominal trajectories are computed as in Appendix A, the polynomial degree is again five, in order to create sufficiently differentiable curves. White noise is added to the measured outputs with a relevant level to the real process measure level. Derivatives are estimated by using a high-gain observer, see 2.5.5 coupled to a low-pass filter to reduce the amplitude of the noise and improve the derivative estimation. Once again a trad-off between time delay and noise filtering is taken into account.

Let us develop the FTC approach dividing it in two different cases.

3.3.3.1 CASE A

FAULT DETECTION AND ISOLATION

Let us only analyze the case when only one set of flat outputs is found, in this system $z_\alpha = [x_1,\ x_3]^T$, for FDI this supposition implies the three hypothesis presented in section 2.5.2. By consequence three residues can be computed as in 3.44.

$$\begin{bmatrix} r^{\alpha}_{1x} \\ r^{\alpha}_{1u} \\ r^{\alpha}_{2u} \end{bmatrix} = \begin{bmatrix} x_{m2} \\ u_{m1} \\ u_{m2} \end{bmatrix} - \begin{bmatrix} \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [0\,1\,0]^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [1\,0]^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [0\,1]^T \end{bmatrix} \tag{3.44}$$

Let us analyze each individual fault. Examining equation (3.44) is straightforward to see that all the right hand of it, is in function of the α set of flat outputs, by consequence if a fault is present in the measure x_{m1} or x_{m3}, all the residues will be impacted, such effect will indicate the presence of a fault but preventing the isolation because both faults will have the same fault signature. The phenomenon is the same with additive and multiplicative fault indistinctly. See Figs. 3.15, 3.16, 3.17 and 3.18.

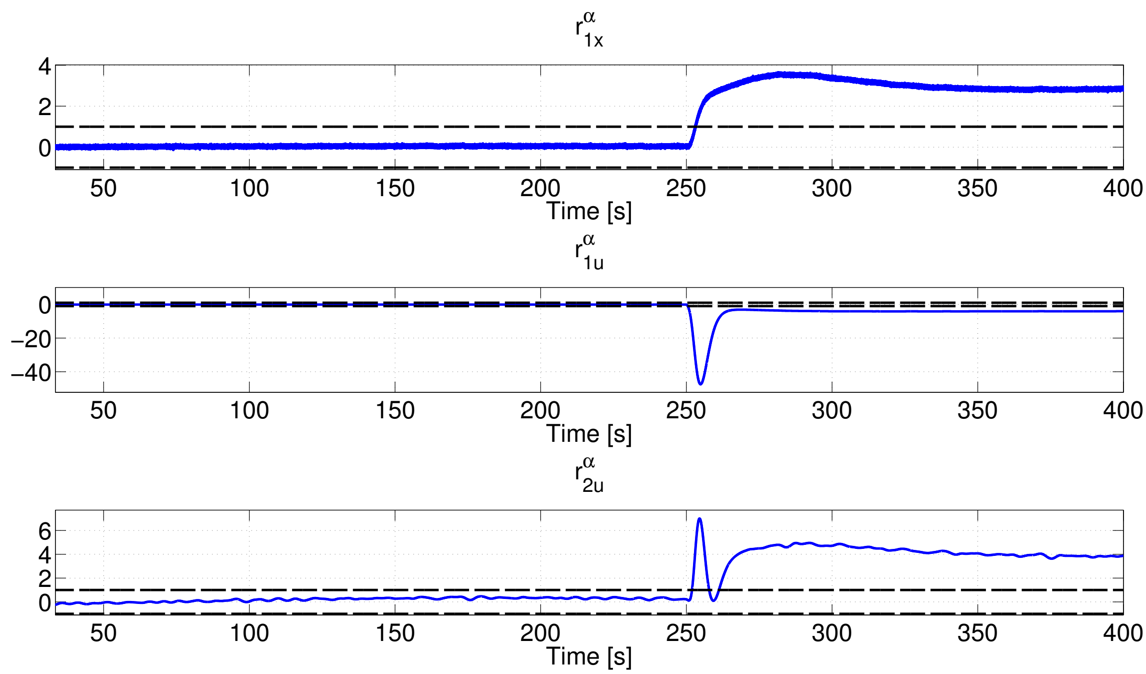

Figure 3.15: Additive fault measure x_1 normalized (z_α set)

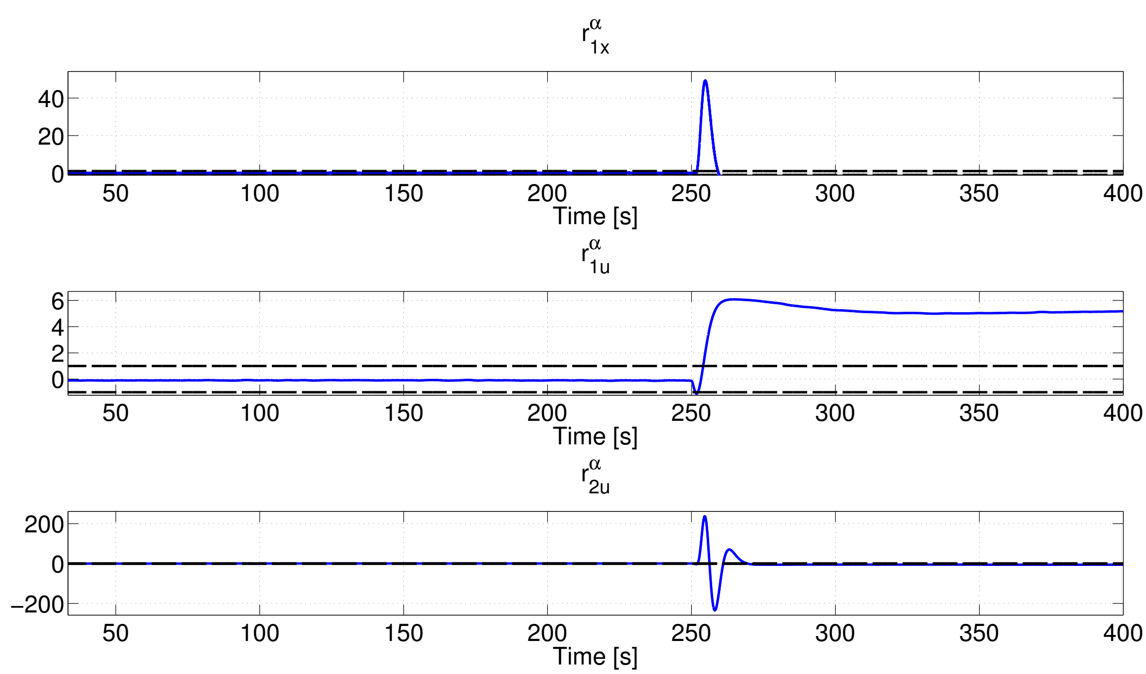

Figure 3.16: Additive fault measure x_3 normalized (z_α set)

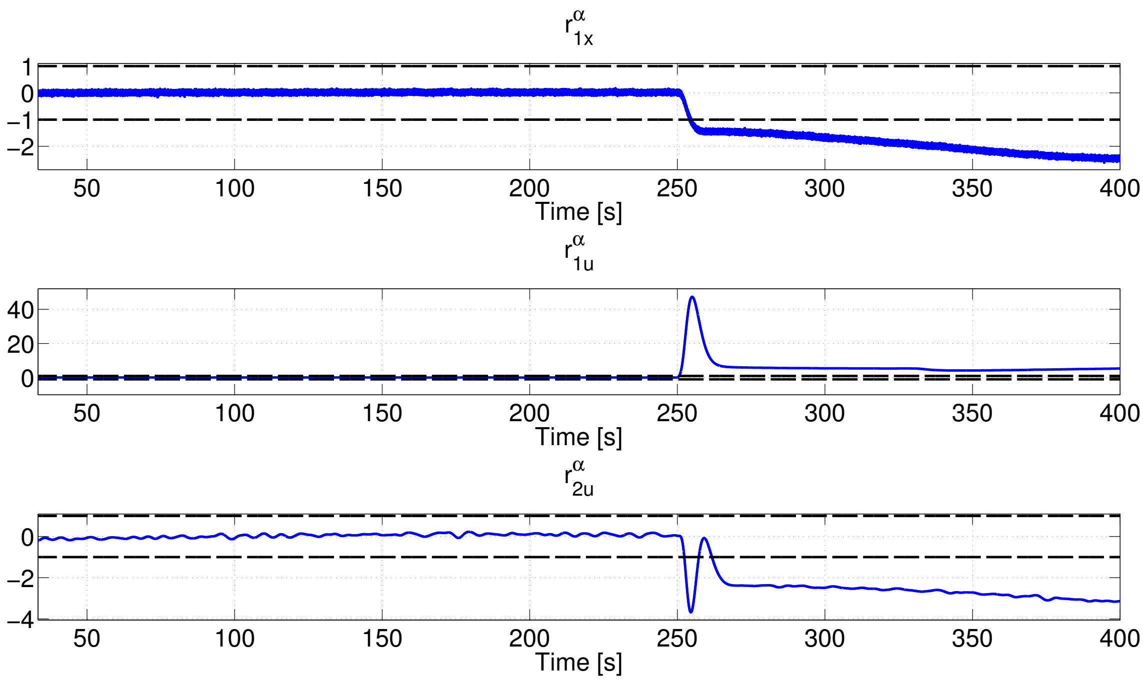

Figure 3.17: Multiplicative fault measure x_1 normalized (z_α set)

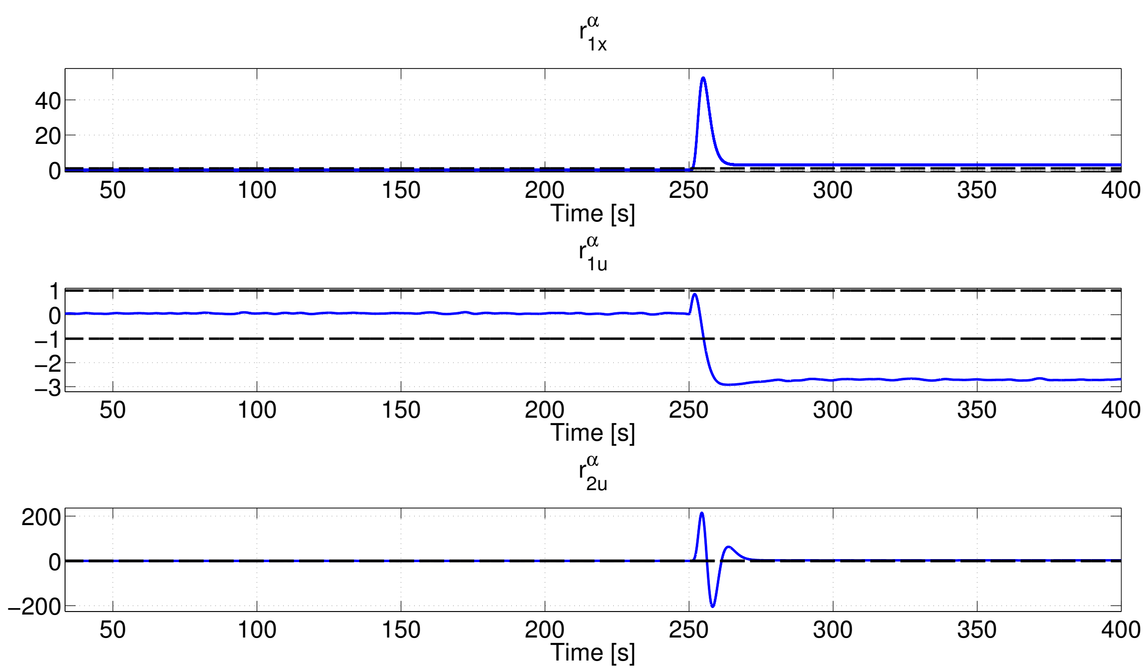

Figure 3.18: Multiplicative fault measure x_3 normalized (z_α set)

Faults affecting actuators will impact r_{1u}^{α}, if the fault is presented in pump u_1 and if the fault affects pump u_2 the residual signal that will change it's amplitude will be r_{2u}^{α}. Fault affecting actuator u_1 is detected and isolated by simply comparing the amplitude of the residual signal versus the threshold amplitude, Figs.3.19 and 3.21. However this strategy in not effective for a fault affecting the actuator u_2, even if the residual signal change the amplitude, it is not big enough to exceed the threshold, Figs. 3.20 and 3.22. Such small change could be detected with another type of decision algorithm. On the other hand sensor fault in high measure of tank number 2 can be detected and isolated, since only r_{1x}^{α} is in function of x_{m2} and by consequence only this residue is affected, providing by this way a particular fault signature. Table 3.4 resume the results. The fault signatures are the same for additive and multiplicative faults.

Fault	r_{1x}^{α}	r_{1u}^{α}	r_{2u}^{α}
F_{x1}	1	1	1
F_{x2}	1	0[4]	1
F_{x3}	1	1	1
F_{u1}	0	1	0
F_{u2}	0	0	0[4]

Table 3.4: Residues matrix Three tanks Case A

[4]This residue is affected but the amplitude is not enough to exceed the threshold.

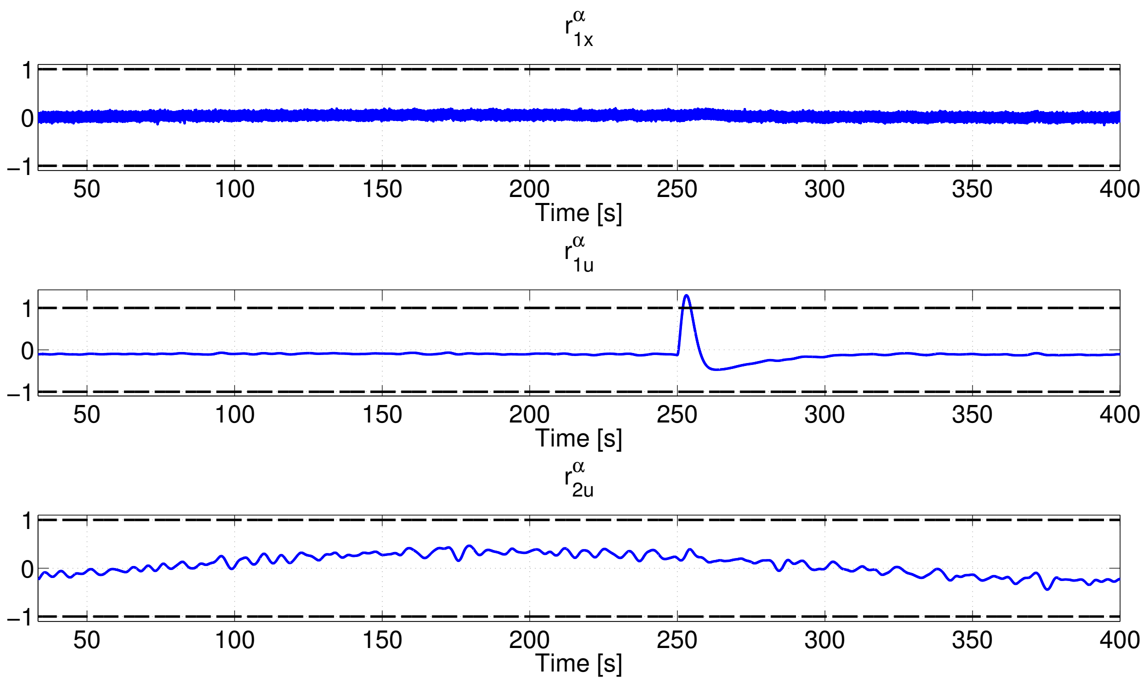

Figure 3.19: Additive fault in flow pump u_1 normalized (z_α set)

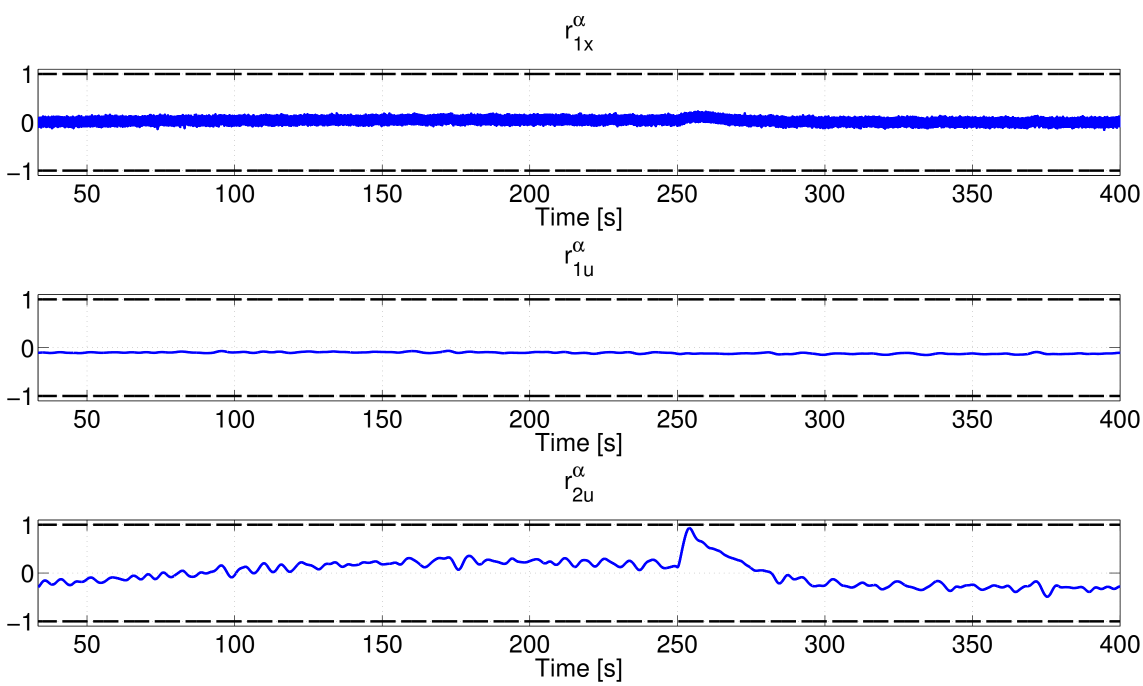

Figure 3.20: Additive fault in flow pump u_2 normalized (z_α set)

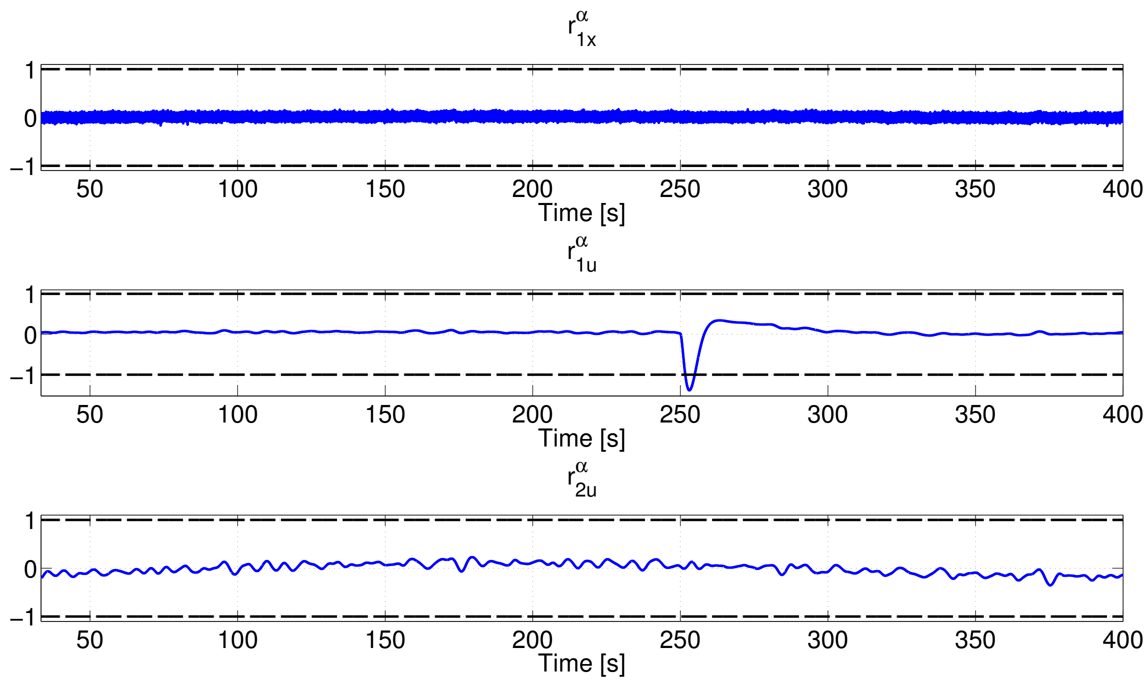

Figure 3.21: Multiplicative fault in flow pump u_1 normalized (z_α set)

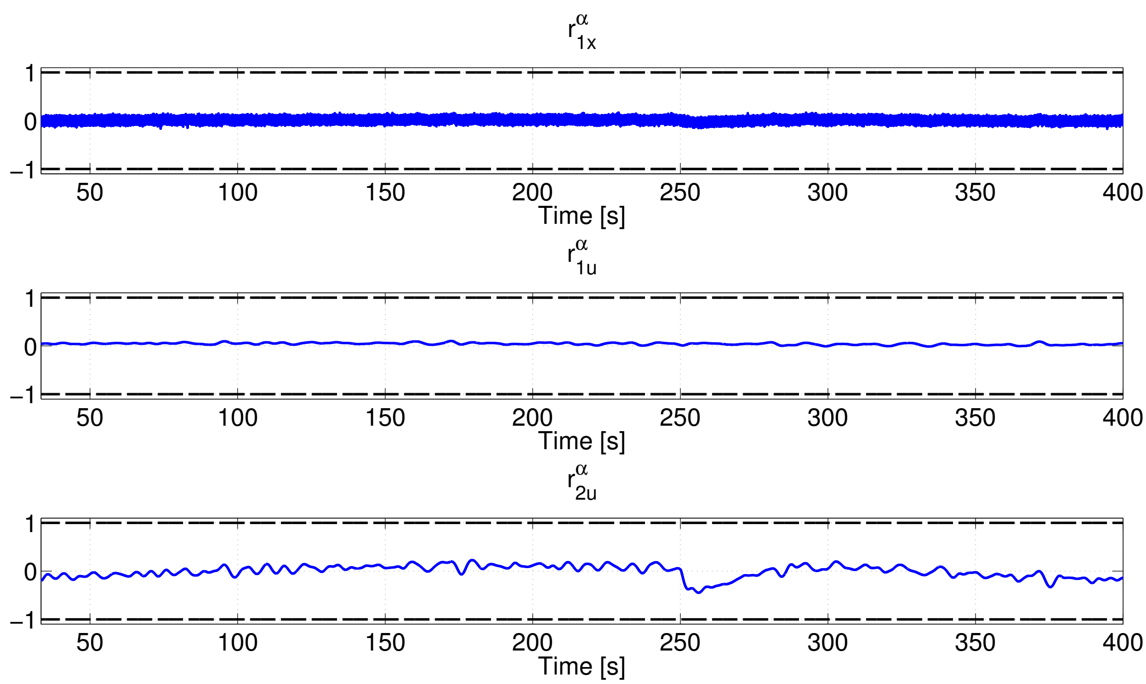

Figure 3.22: Multiplicative fault in flow pump u_2 normalized (z_α set)

Figures 3.23 and 3.24 shows the three residual signals obtained when a fault affects the high measure of tank number two. Such behavior is explained by the directly relation of the r^{α}_{1x} and the measure coming from sensor of the tank number two. Since this tank is directly related to the pump number two, the controller tries to compensate the fault, such reaction impact the residual which depends on pump two. Residual r^{α}_{1u} is affected because the pump number one tries to compensate the fault, however this is not directly related, by consequence the amplitude is not enough to exceeds the threshold and the fault can be detected but cannot be isolated. Such effect could be avoided by using a more sophisticated FDI decision algorithm.

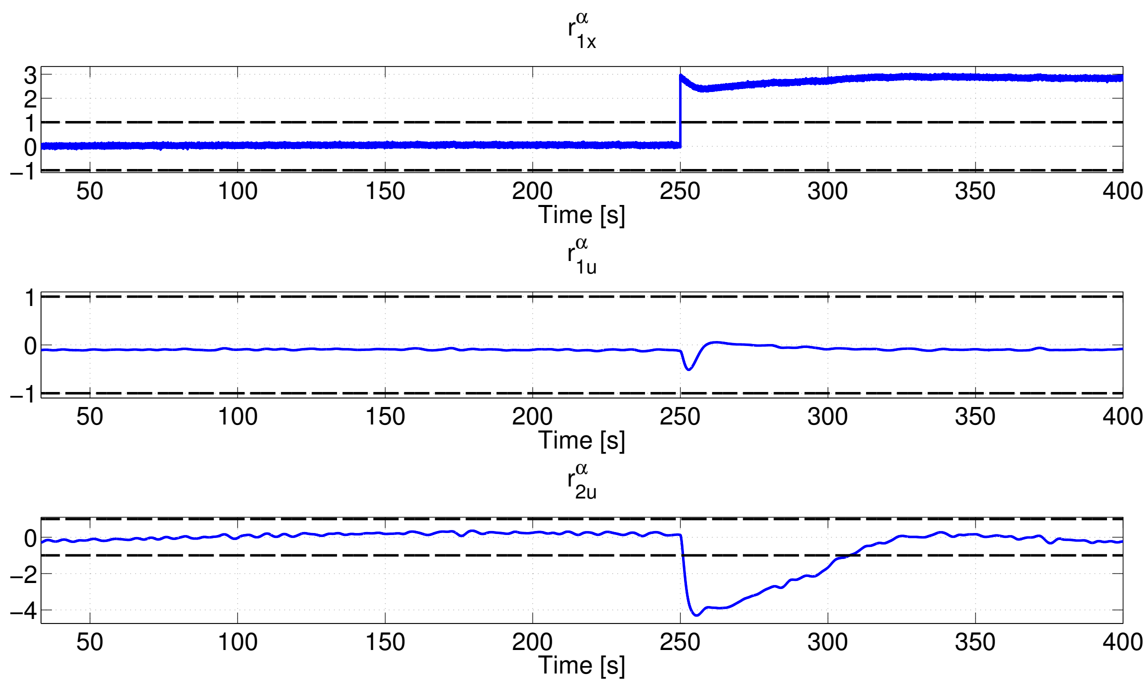

Figure 3.23: Additive fault measure x_2 normalized (z_α set)

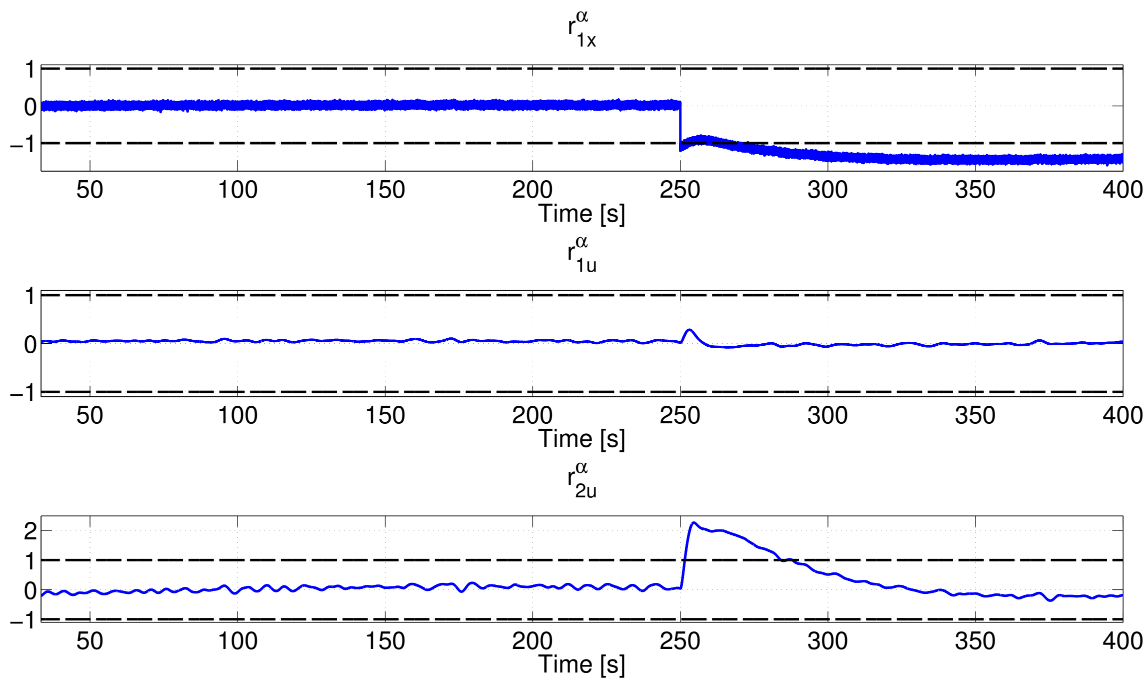

Figure 3.24: Multiplicative fault measure x_2 normalized (z_α set)

CONTROL RECONFIGURATION

The number of redundant signals and by consequence the number of reconfigurable faults could be obtained using the equation (2.38), so, the number of redundant signals available is $(1) * (3 - 2) = 1$. The only signal that could be estimated is the one representing the high measure of tank number 2. The estimated signal is obtained using the expression x_2^α in the equation (3.38), thanks to that expression only depends on the z_α vector the estimated signal $\hat{x_2}$ is fault-free, hence $\hat{x_2}$ substitutes the faulty signal x_{m2} in the state feedback. Figures 3.25 and 3.26 presents the comparison of final positions with and without reconfiguration. It is clearly to see that if the signal is not reconfigured the system does not reach the desired final value.

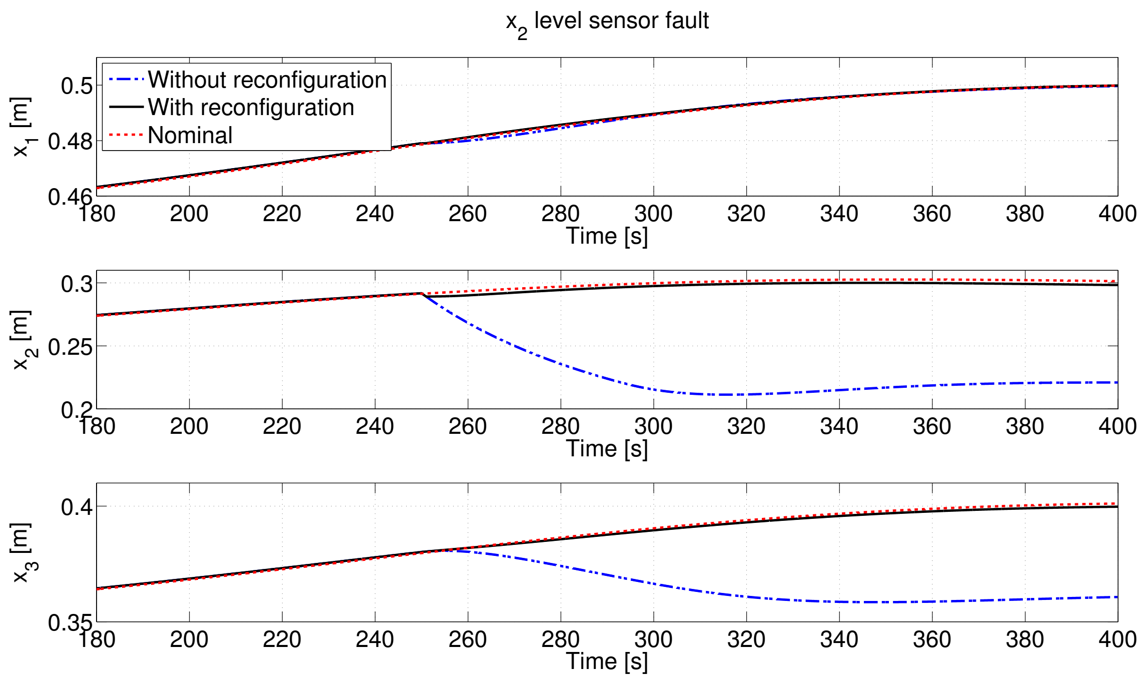

Figure 3.25: Reconfiguration after additive fault in x_2

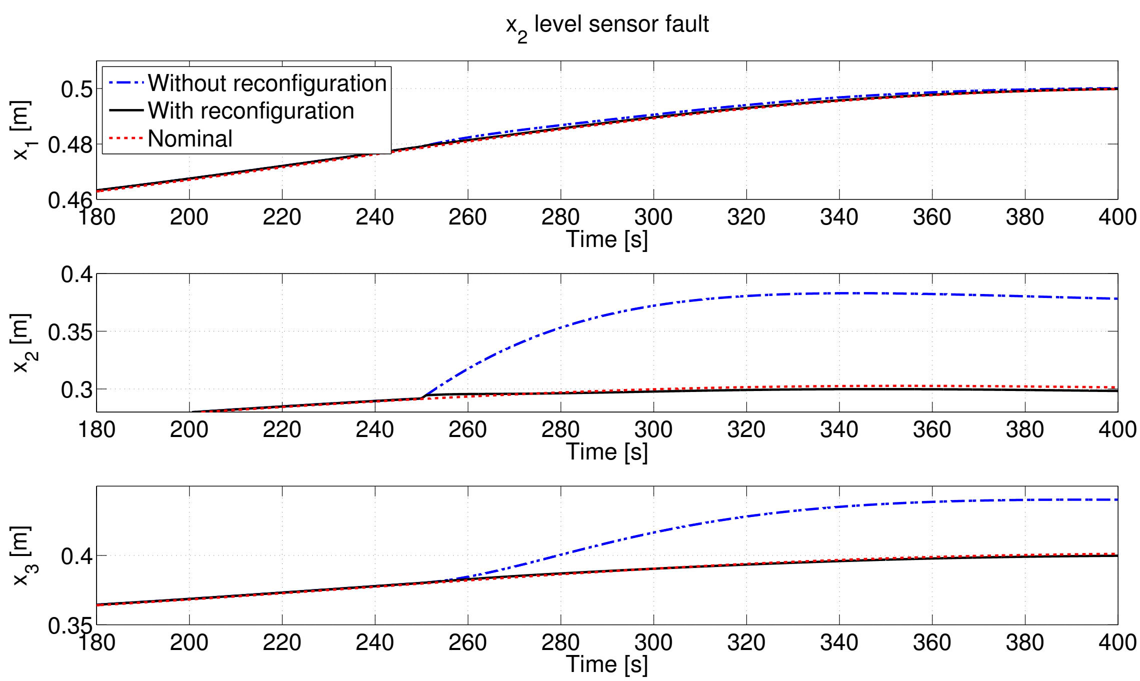

Figure 3.26: Reconfiguration after multiplicative fault in x_2

3.3.3.2 CASE B

FAULT DETECTION AND ISOLATION

Using the two available set of flat outputs z_α and z_β the number of residues is duplicated, for this system the number of residuals is increased to 6 $(n+n)$. The residual signals are computed as in the previous case and are presented in equation (3.45).

$$\begin{bmatrix} r_{1x}^{\alpha} \\ r_{1u}^{\alpha} \\ r_{2u}^{\alpha} \\ r_{1x}^{\beta} \\ r_{1u}^{\beta} \\ r_{2u}^{\beta} \end{bmatrix} = \begin{bmatrix} x_{m2} \\ u_{m1} \\ u_{m2} \\ x_{m1} \\ u_{m1} \\ u_{m2} \end{bmatrix} - \begin{bmatrix} \phi_{\alpha x}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [0\,1\,0]^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [1\,0]^T \\ \phi_{\alpha u}(z_{\alpha 1}, \dot{z}_{\alpha 1}, z_{\alpha 2}, \dot{z}_{\alpha 2})\ [0\,1]^T \\ \phi_{\beta x}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ [1\,0\,0]^T \\ \phi_{\beta u}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ [1\,0]^T \\ \phi_{\beta u}(z_{\beta 1}, \dot{z}_{\beta 1}, z_{\beta 2}, \dot{z}_{\beta 2})\ [0\,1]^T \end{bmatrix} \tag{3.45}$$

The fact of increasing the number of residual signals help to improve the FDI stage. Let us present the framework of each fault. Actuator fault u_1 is as in the case A detected and isolated by simply comparing the residuals amplitude versus the threshold, see Figs. 3.27 and 3.29. Once again u_2 fault present one residue which depends of the u_{m2} but this one does not exceed the threshold, however this time thanks to the second vector z_β an additional residue r_{2u}^{β} is present, such vector exceeds the threshold, this behavior provide an individual fault signature for u_2 fault, see Table 3.5 and Figs. 3.28 and 3.30.

For sensor faults, for instance a fault affecting high measure of tank one will affect the three residues in the upper part of the right side in the equation (3.45) because for this residual signals x_1 is a flat output. Besides the residual signal r_{1x}^{β} depends on the measure x_{m1}, by consequence it will be affected too. see Figs. 3.31 and 3.32. Residual signal r_{2u}^{β} is affected because the pump number two reacts to the fault as a reflect of the pump to counteract the fault, however the amplitude is not enough to exceed the threshold.

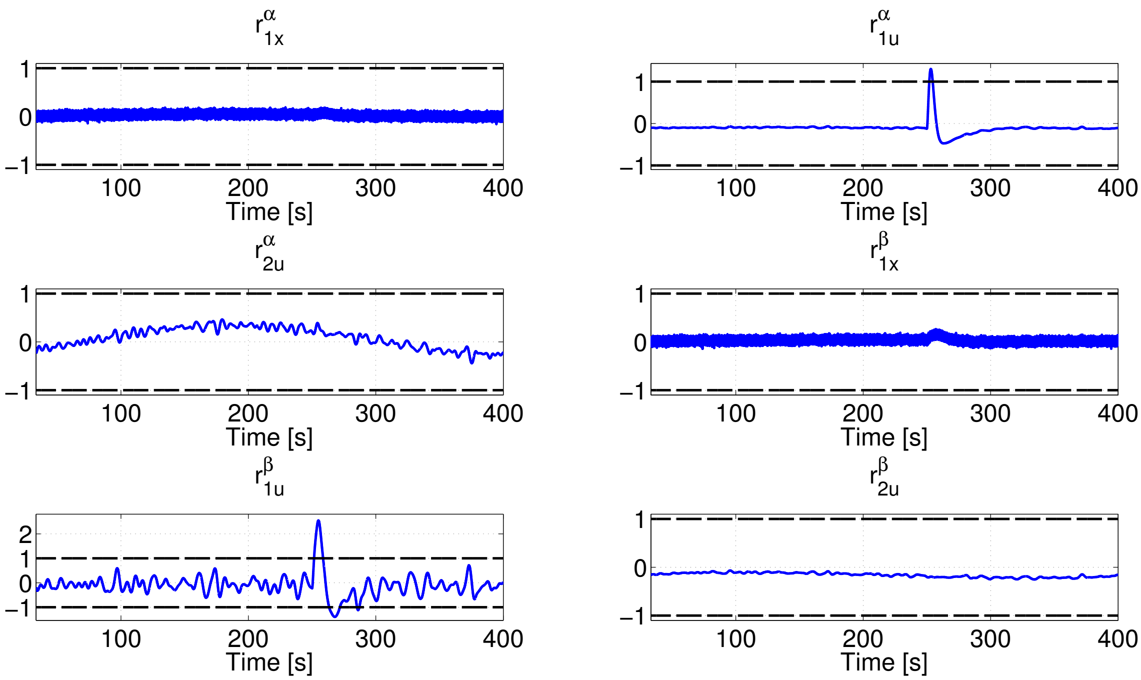

Figure 3.27: Additive fault in flow pump u_1 normalized (z_α and z_β set)

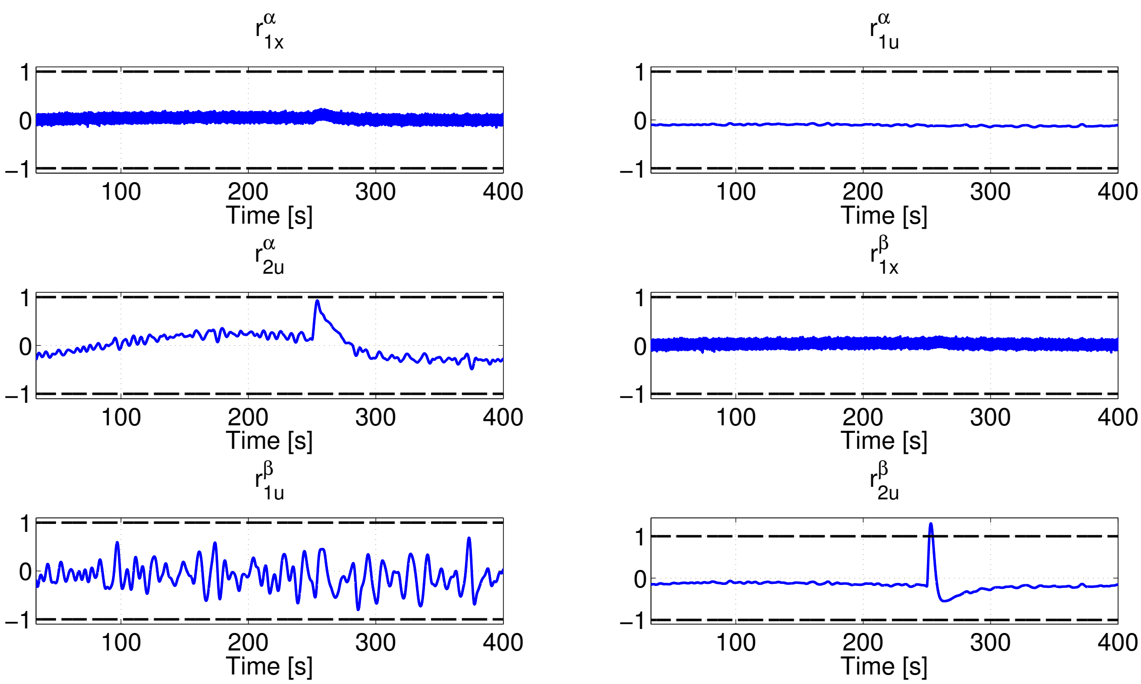

Figure 3.28: Additive fault in flow pump u_2 normalized (z_α and z_β set)

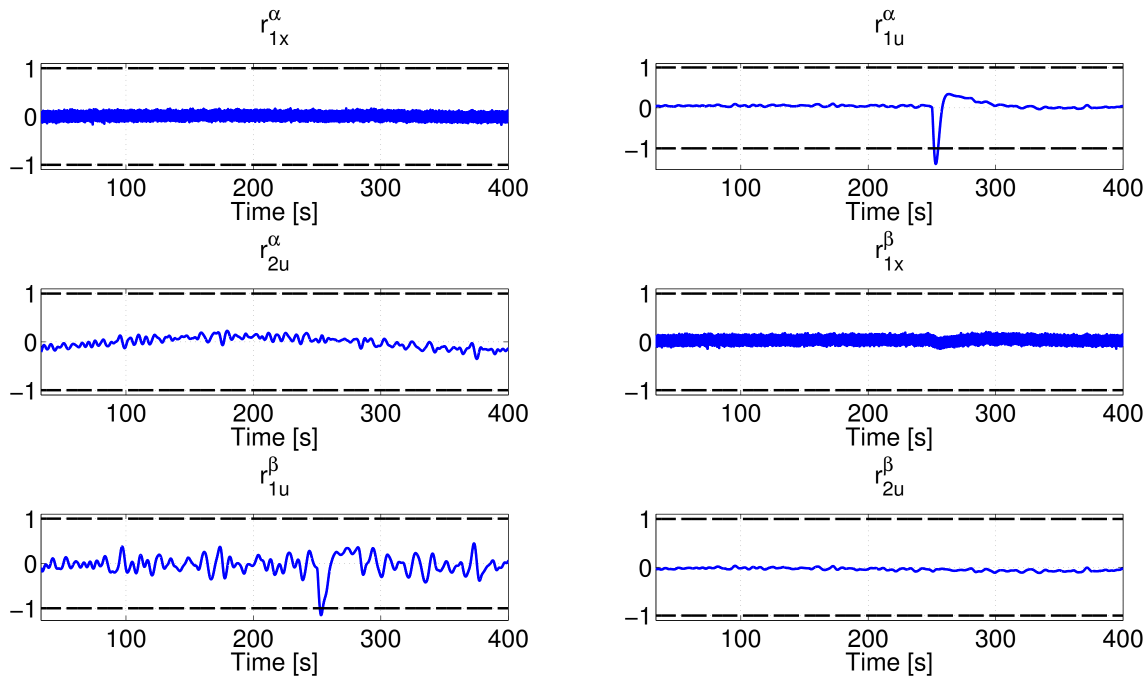

Figure 3.29: Multiplicative fault in flow pump u_1 normalized (z_α and z_β set)

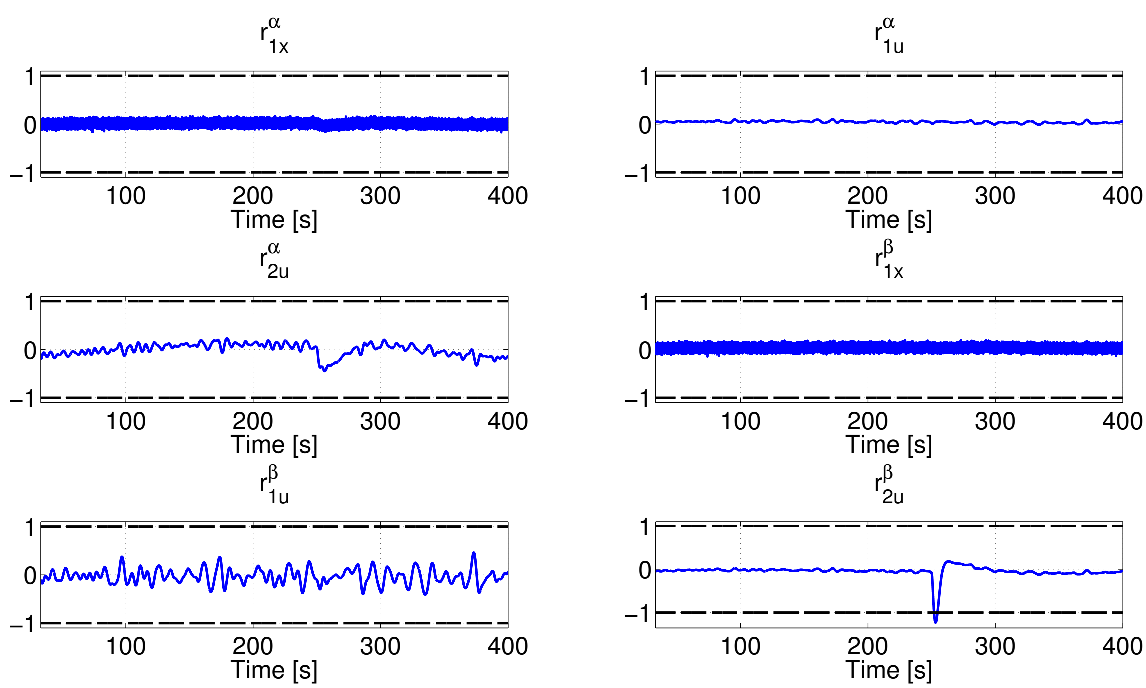

Figure 3.30: Multiplicative fault in flow pump u_2 normalized (z_α and z_β set)

Fault	r^{α}_{1x}	r^{α}_{1u}	r^{α}_{2u}	r^{β}_{1x}	r^{β}_{1u}	r^{β}_{2u}
F_{x1}	1	1	1	1	1	0[5]
F_{x2}	1	0[5]	1	1	1	1
F_{x3}	1	1	1	1	1	1
F_{u1}	0	1	0	0	1	0
F_{u2}	0	0	0[5]	0	0	1

Table 3.5: Residues matrix Three tank Case B

[5]This residue is affected but the amplitude is not enough to exceed the threshold.

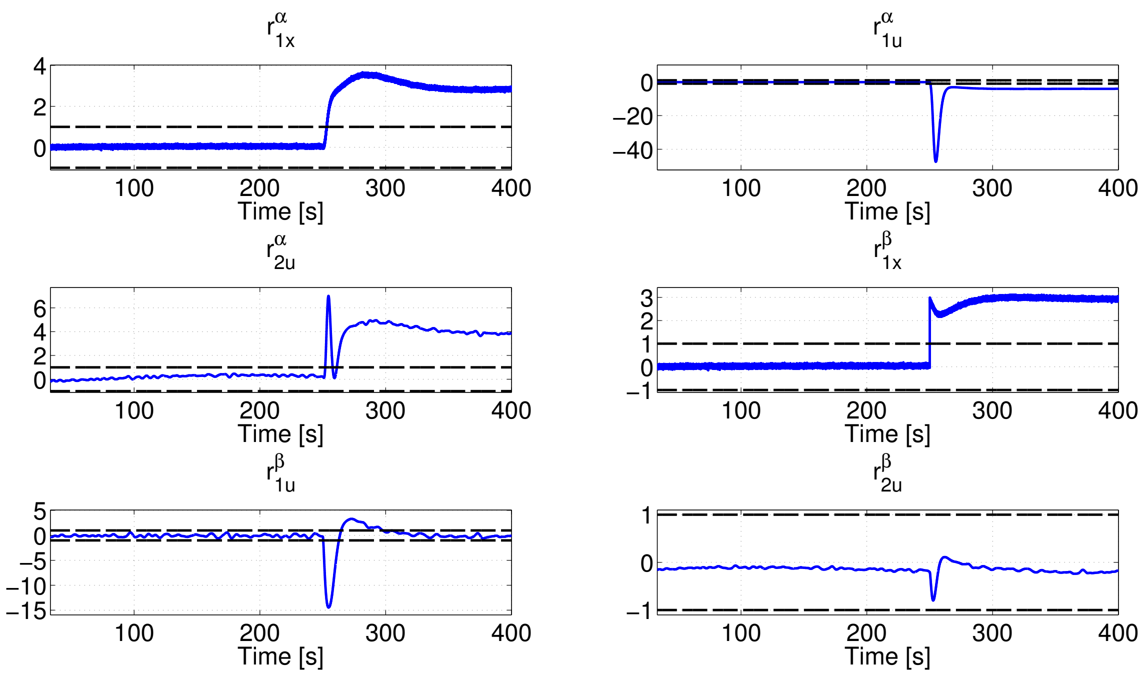

Figure 3.31: Additive fault measure x_1 normalized (z_α and z_β set)

If the fault affects the high measure of tank number two x_{m2} the affected residues depend on the z_β vector, by consequence the residues r^{β}_{1x}, r^{β}_{1u} and r^{β}_{2u} are affected, the residual r^{α}_{1x} is affected too because the presence of x_{m2}. This time the residual signal affected as consequence of the closed loop is r^{α}_{1u}. See Figs. 3.33 and 3.34.

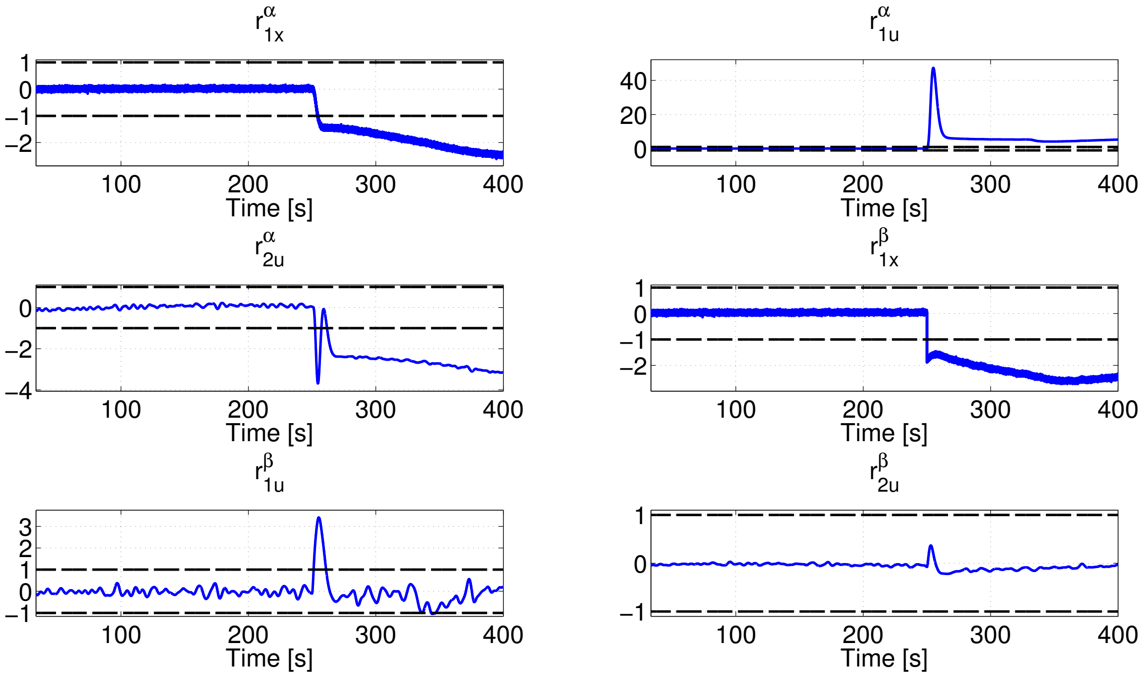

Figure 3.32: Multiplicative fault measure x_1 normalized (z_α and z_β set)

Fault in x_{m3} is an special case, because the state x_3 is part of both flat output vectors, by consequence the six residues will be affected. However this is the only framework in which every residue change its behavior, so, the fault can be detected and isolated. See Figs. 3.35 and 3.36. Table 3.5 resumes the different fault signatures.

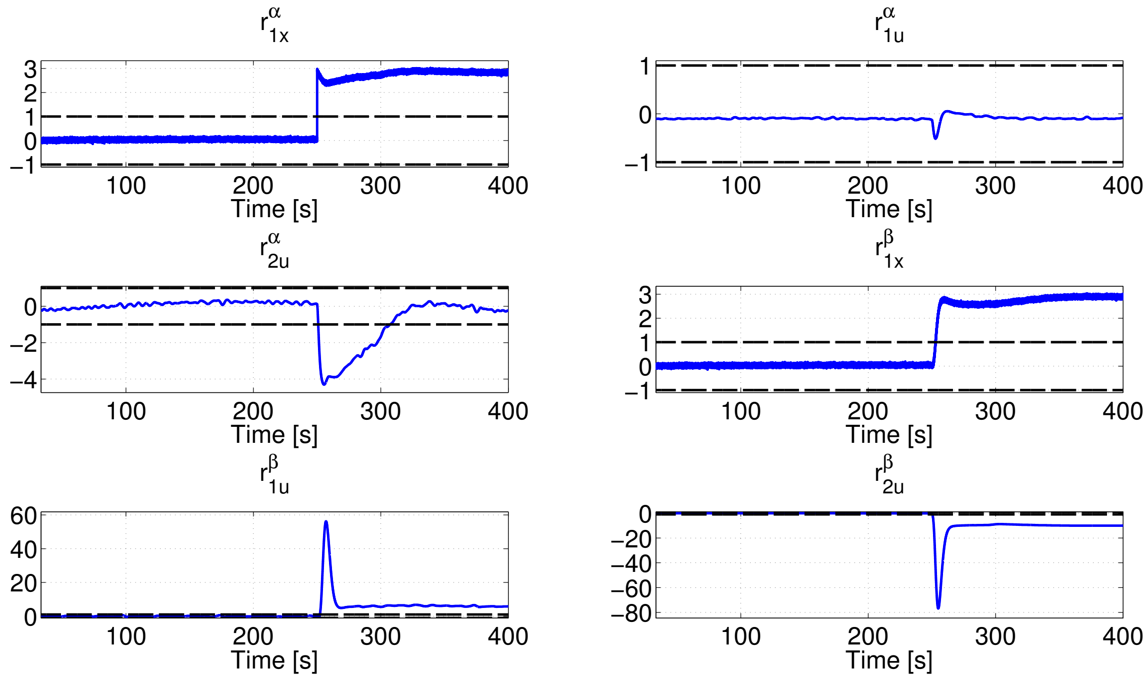

Figure 3.33: Additive fault measure x_2 normalized (z_α and z_β set)

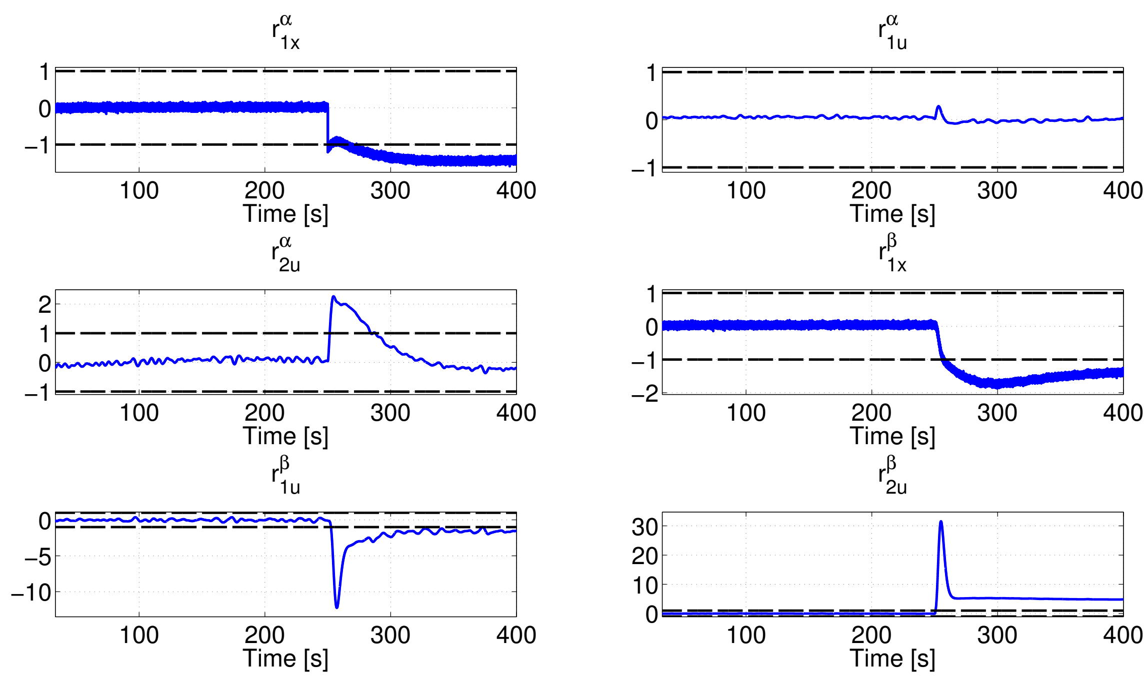

Figure 3.34: Multiplicative fault measure x_2 normalized (z_α and z_β set)

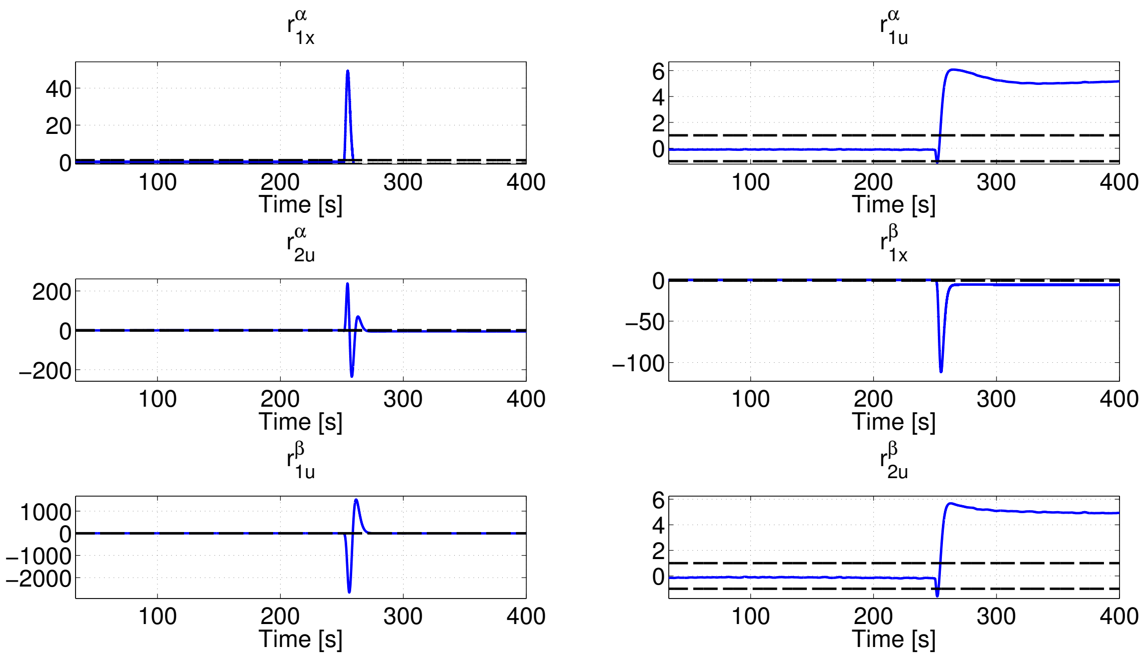

Figure 3.35: Additive fault measure x_3 normalized (z_α and z_β set)

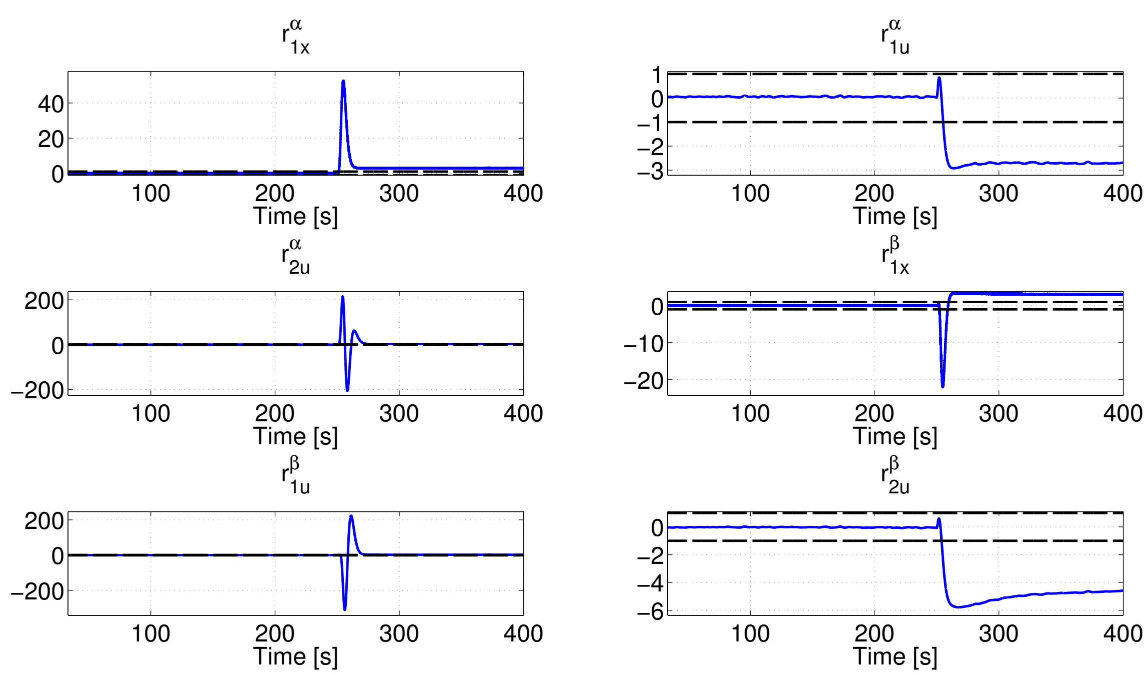

Figure 3.36: Multiplicative fault measure x_3 normalized (z_α and z_β set)

CONTROL RECONFIGURATION

As a direct consequence of obtaining a second set of flat outputs, not only the number of residual signals is increased, the number of redundant signals is augmented as well. This time the equation (2.38) becomes (2)*(3-2)=2, since the system has 3 states, full reconfiguration is not possible. Observing in detail the results of the FDI stage and the flat output vectors it is straightforward to see that the state x_3 triggers all the residual signals because it is part of both flat output vectors. By consequence it is impossible to compute a redundant fault-free version of it. Such effect prevents the reconfiguration after a fault on x_3. This fault is not considered.

The two redundant signals available to accomplish the FTC approach are as in case A, the state x_2, the additionally set of flat outputs z_β provides a fault-free version of x_1. The reconfiguration is obtained in the same manner that the case A. Figs. 3.37 and 3.38 depicts the trajectories of the outputs with and without reconfiguration. Once again it exists a remarkable difference between the trajectories with and without reconfiguration. Such results proof the efficiency of the proposed approach.

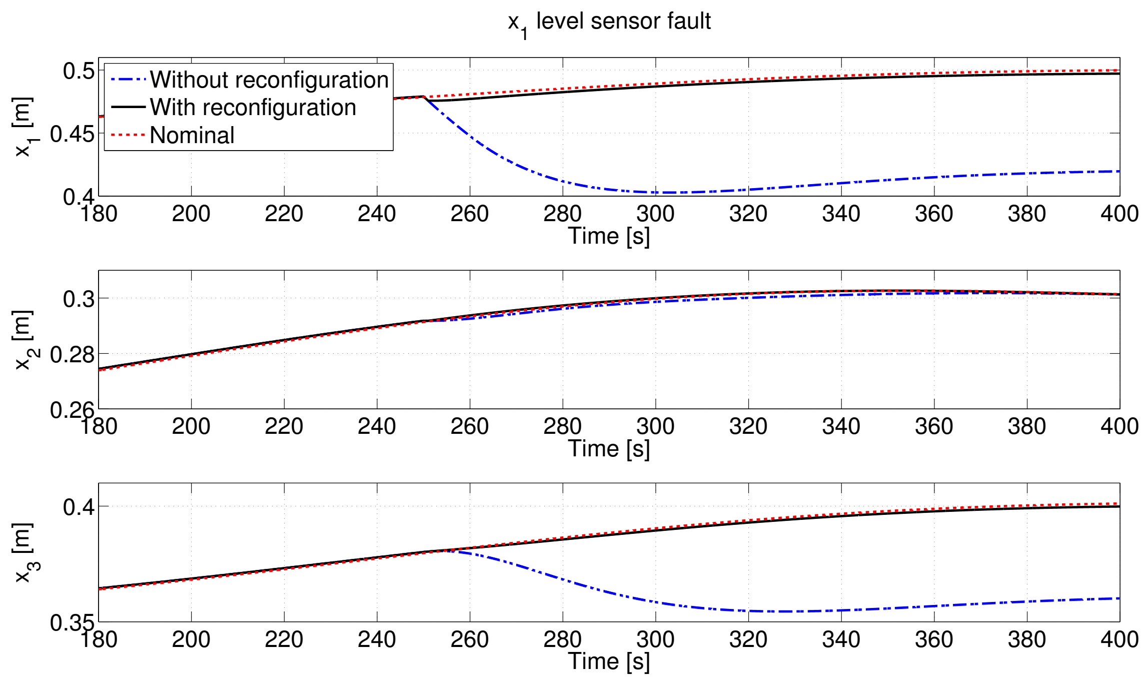

Figure 3.37: Reconfiguration after additive fault in x_1

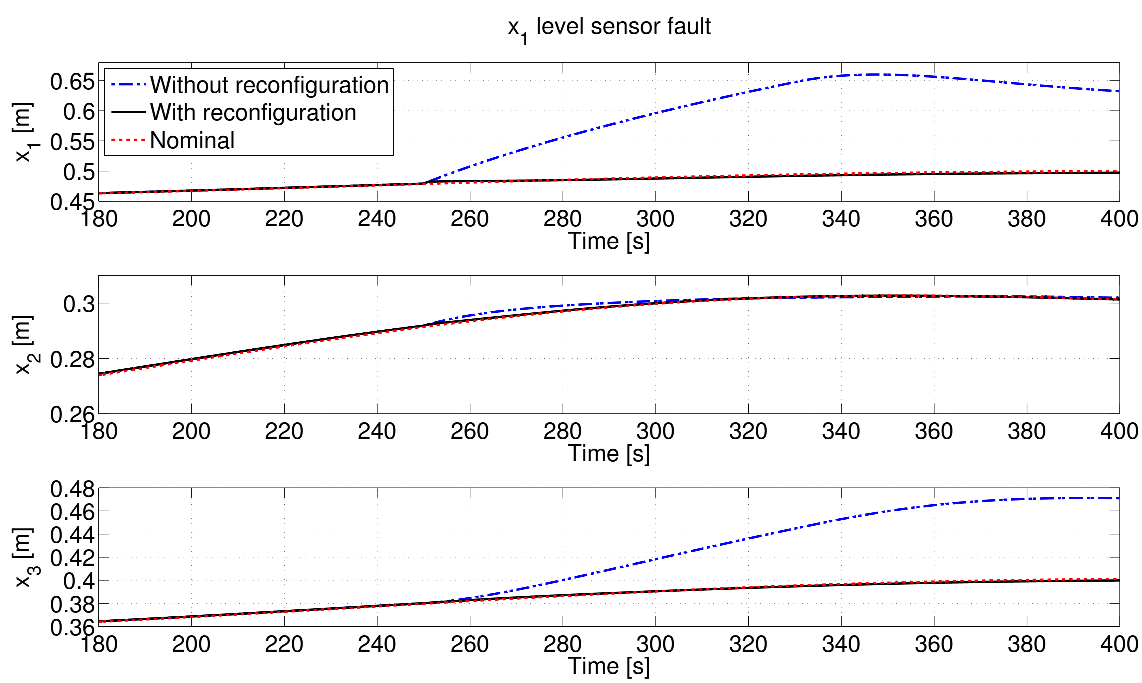

Figure 3.38: Reconfiguration after multiplicative fault in x_1

3.4 CONCLUSION

The proposed approach presented in chapter 2 is tested in this chapter. Two different systems were selected, a UAV quadrotor and a three tank system. For the quadrotor only one set of flat outputs is found, however control reconfiguration can be done partially. For the three tank process two sets fulfilling the conditions to exploit at maximum the technique are found, this has as consequence the fact that any fault may be detected and isolated. Fault detection and isolation is carried out by simply comparing the residual amplitude and a pre-defined fix threshold. For this specific system, even if in one fault case one of the residues is affected but does not exceed the threshold, every single fault can be detected and isolated. This problem can be avoided by changing threshold-based isolation mechanism for a more adequate one [**?**].

Control reconfiguration is carried out by simply switching between the fault and the unfaulty measure, since our approach uses the non-uniqueness property of the flat vector. Unfaulty signals to reconfigure all the fault measures can only be obtained if every element of the flat output vectors is differentially coupled and algebraically independent, and if the flat outputs are the state of the system or a linear combination of them.

Additionally, for additive faults the amplitude can be estimated by simply subtracting the faulty version from the fault-free one. This information could be useful in the future, in order to plan an optimal trajectory after failure.

Final conclusion

In this thesis, a flatness-based FTC approach is presented, such approach could be applied to nonlinear and linear systems, such systems has to be differentially flat, this property permits to write every system state and every control input as function of some internal variables, so-called flat outputs. The FTC approach take advantage of the non-uniqueness property of the flat output vector, in fact if there exists at least two set of flat outputs and at least one of their internal elements are algerbarically independents, the FDI could be improved in a considerable manner. This operation is enhanced because if the supposition presented before is verified the number of residual signals is augmented, by consequence the probabilities to obtain a different fault signature for each fault augments too.

Real applicability is verified in two different nonlinear systems, a quadrotor UAV, for this system only one set of flat outputs could be found, however thanks that the internal decoupling present in this system each single sensor fault could be detected and isolated. Additionally thanks that the properties of the flat systems, fault-free references of the system states are available. Such signals are used to change the controller reference. This action hide the fault to the controller, and by consequence the system is not impacted by the fault.

The second system is a classical three tank process, contrarily to the UAV, this time two set of flat outputs are found. As consequence the number of residues is augmented and every single fault could be detected. Additionally the fault-free references could be used for reconfiguration.

The derivatives needed to compute the unfaulty references are computed with high-gain observers coupled with a low pass filter in both systems. However the internal parameters of each block are tuned according to the system, for instance the delay introduced for the low-pass filter is not a key parameter in the three tank process, because the dynamic of this system is slow. By consequence the filter is designed to eliminate noise and does not take into account the resulting time delay. On the other hand the dynamic of the UAV is faster, so the design of the filter needs special attention, in fact if the cut-off frequency is big the time delay induced to the estimates will be considerable, however if the cut-off frequency is low the time delay will decrease but the noise will increase. As consequence a trade-off needs to be found.

Even if the technique shows to be effective to counteract the fault effect for both systems, the proposed approach has some limitations. The fact that the flat outputs has to be system states or linear combination of them could reduce the applicability. Another important point is the fact that because the technique is based in Flatness it becomes necessary to compute the time derivatives of noisy signals, which could be mount in difficulty when the time derivatives mount in order.

FUTURE WORK

The chapter number three summarized the results presented in this dissertation. The proposed FTC technique and their applications to improve the fault detection and reconfiguration of nonlinear systems were described. Besides the admired features of the proposed methods, there is a room for further improvements. In below, we outline a few possible directions for possible extension of the work.

- Obtaining two set of flat outputs could be a hard task. A future direction of this work could consist in develop an automatic algorithm to do this computations or at least present the necessary conditions, in which it exists two or more sets of flat outputs.

- The FDI decision is taken by a simply fixed algorithm, even if technique is effective, in the next work a more sophisticated decision algorithm could be tested.

- Reconfiguration is carried by changing the faulty signal for an estimated reference. This change is carried out by means of a switch, such action could produce instability. Future work will be pointed to study this phenomenon and give solutions to avoid stability.

- Another pending issue is related to the real application to both nonlinear systems. Additionally for the UAV most of the dynamics are neglected, in order to apply the proposed approach to a real UAV the nonlinear model presented in this manuscript could be change for a more accurate one.

- Reconfiguration shows its applicability to the UAV, however there is some limitations regarding the fault size, another interesting work could be investigate the restructuring of the control loop.

ABBREVIATIONS AND NOTATIONS

Abbreviations

Abbreviation	Meaning
AFTCS	Active Fault Tolerant Control Systems.
DOF	Degrees Of Freedom.
EKF	Extended Kalman Filter.
ELS	Extended Least Squares.
FDI	Fault Detection and Isolation.
HOT	High Order Terms.
LQ	Linear Quadratic.
LS	Least Squares.
MPC	Model Predictive Control.
NIO	Nonlinear Identity Observer.
PID	Proportional Integral Derivative.
RLS	Recursive Least Squares.
UAV	Unmanned Aerial Vehicle.

BIBLIOGRAPHY

[1] ADJALLAH, K., D. MAQUIN and J. RAGOT, «Non-linear observer-based fault detection», in *Control Applications, 1994., Proceedings of the Third IEEE Conference on*, IEEE, pages. 1115–1120, 1994.

[2] ANTRITTER, F., B. MÜLLER and J. DEUTSCHER, «Tracking control for nonlinear flat systems by linear dynamic output feedback», *Proceedings NOLCOS 2004, Stuttgart*, 2004.

[3] BENOSMAN, M., «Passive Fault Tolerant Control, Robust control, Theory and applications», , 2011.

[4] BODSON, M. and J. E. GROSZKIEWICZ, «Multivariable adaptive algorithms for reconfigurable flight control», *Control Systems Technology, IEEE Transactions on*, **5**(2), pages. 217–229, 1997.

[5] CAGLAYAN, A., S. ALLEN and K. WEHMULLER, «Evaluation of a second generation reconfiguration strategy for aircraft flight control systems subjected to actuator failure/surface damage», in *Aerospace and Electronics Conference, 1988. NAECON 1988., Proceedings of the IEEE 1988 National*, IEEE, pages. 520–529, 1988.

[6] CASBEER, D. W., D. B. KINGSTON, R. W. BEARD and T. W. MCLAIN, «Cooperative forest fire surveillance using a team of small unmanned air vehicles», *International Journal of Systems Science*, **37**(6), pages. 351–360, 2006.

[7] CASTILLO, P., A. DZUL and R. LOZANO, «Real-time stabilization and tracking of a four-rotor mini rotorcraft», *Control Systems Technology, IEEE Transactions on*, **12**(4), pages. 510–516, 2004.

[8] CAZAURANG, F. and L. LAVIGNE, «Satellite path planning by flatness approach», *International Review of Aerospace Engineering*, **2**(3), pages. 123–132, 2009.

[9] CHAMSEDDINE, A., Y. ZHANG, C. A. RABBATH, C. JOIN and D. THEILLIOL, «Flatness-Based Trajectory Planning/Replanning for a Quadrotor Unmanned Aerial Vehicle», *Aerospace and Electronic Systems, IEEE Transactions on*, **48**(4), pages. 2832–2848, 2012.

[10] CHARLET, B., J. LÉVINE and R. MARINO, «Sufficient conditions for dynamic state feedback linearization», *SIAM journal on control and optimization*, **29**(1), pages. 38–57, 1991.

[11] CHEN, J. and R. J. PATTON, *Robust model-based fault diagnosis for dynamic systems*, Kluwer academic publishers, 1999.

[12] CHOW, E. and A. WILLSKY, «Analytical redundancy and the design of robust failure detection systems», *Automatic Control, IEEE Transactions on*, **29**(7), pages. 603–614, 1984.

[13] CLARK, R. N., D. C. FOSTH and V. M. WALTON, «Detecting instrument malfunctions in control systems», *Aerospace and Electronic Systems, IEEE Transactions on*, (4), pages. 465–473, 1975.

[14] COWLING, I. D., O. A. YAKIMENKO, J. F. WHIDBORNE and A. K. COOKE, «A prototype of an autonomous controller for a quadrotor UAV», in *European Control Conference*, pages. 1–8, 2007.

[15] CRASSIDIS, J. L. and J. L. JUNKINS, *Optimal Estimation of Dynamic Systems*, Chapman and Hall/CRC, 2004.

[16] DALMASSO, I., I. GALLETTI, R. GIULIANO and F. MAZZENGA, «WiMAX networks for emergency management based on UAVs», in *Satellite Telecommunications (ESTEL), 2012 IEEE First AESS European Conference on*, IEEE, pages. 1–6, 2012.

[17] DEVASIA, S. and B. PADEN, «Exact output tracking for nonlinear time-varying systems», in *Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on*, volume 3, IEEE, pages. 2346–2355, 1994.

[18] DING, S. X., *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*, Springer, 2008.

[19] EDWARDS, C., S. K. SPURGEON and R. J. PATTON, «Sliding mode observers for fault detection and isolation», *Automatica*, **36**(4), pages. 541–553, 2000.

[20] FINN, R. L. and D. WRIGHT, «Unmanned aircraft systems: Surveillance, ethics and privacy in civil applications», *Computer Law & Security Review*, **28**(2), pages. 184–194, 2012.

[21] FLIESS, M., C. JOIN, M. MBOUP and H. SIRA-RAMÍREZ, «Compression différentielle de transitoires bruités», *Comptes Rendus Mathematique*, **339**(11), pages. 821–826, 2004.

[22] FLIESS, M., C. JOIN, M. MBOUP, H. SIRA-RAMIREZ *et al.*, «Analyse et représentation de signaux transitoires: application à la compression, au débruitage et à la détection de ruptures», in *GRETSI*, 2005.

[23] FLIESS, M., J. LÉVINE, P. MARTIN and P. ROUCHON, «Flatness and defect of non-linear systems: introductory theory and examples», *International journal of control*, **61**(6), pages. 1327–1361, 1995.

[24] FRANK, P., G. SCHRIER and E. A. GARCIA, «Nonlinear observers for fault detection and isolation», in *New Directions in nonlinear observer design*, Springer, pages. 399–422, 1999.

[25] FRANK, P. M. and M. BLANKE, «Fault diagnosis and fault-tolerant control», *Control Systems, Robotics and Automation, from Encyclopedia of Life Support Systems (EOLSS), developed under the Auspices of the UNESCO*, 2001.

[26] GAO, Z. and P. J. ANTSAKLIS, «Stability of the pseudo-inverse method for reconfigurable control systems», *International Journal of Control*, **53**(3), pages. 717–729, 1991.

[27] GAO, Z. and P. J. ANTSAKLIS, «Reconfigurable control system design via perfect model following», *International Journal of Control*, **56**(4), pages. 783–798, 1992.

[28] GERTLER, J. and D. SINGER, «A new structural framework for parity equation-based failure detection and isolation», *Automatica*, **26**(2), pages. 381–388, 1990.

[29] GOERZEN, C., Z. KONG and B. METTLER, «A survey of motion planning algorithms from the perspective of autonomous UAV guidance», *Journal of Intelligent and Robotic Systems*, **57**(1-4), pages. 65–100, 2010.

[30] GOH, K. B., S. K. SPURGEON and N. B. JONES, «Fault diagnostics using sliding mode techniques», *Control Engineering Practice*, **10**(2), pages. 207–217, 2002.

[31] HENGY, D. and P. FRANK, «Component failure detection via nonlinear state observers», in *Proc. of IFAC Workshop on Fault Detection and Safety in Chemical Plants*, pages. 153–157, 1986.

[32] HORN, J., J. BAMBERGER, P. MICHAU and S. PINDL, «Flatness-based clutch control for automated manual transmissions», *Control Engineering Practice*, **11**(12), pages. 1353–1359, 2003.

[33] HUGENHOLTZ, C. H., B. J. MOORMAN, K. RIDDELL and K. WHITEHEAD, «Small unmanned aircraft systems for remote sensing and earth science research», *Eos, Transactions American Geophysical Union*, **93**(25), pages. 236–236, 2012.

[34] ISERMANN, R., *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*, Springer, 2006.

[35] JOIN, C., H. SIRA-RAMIREZ and M. FLIESS, «Control of an uncertain three-tank-system via on-line parameter identification and fault detection», in *Proc. 16th IFAC World Congress on Automatic Control, Prague*, 2005.

[36] KANTNER, M., B. BODENHEIMER, P. BENDOTTI and R. M. MURRAY, «An experimental comparison of controllers for a vectored thrust, ducted fan engine», in *American Control Conference, 1995. Proceedings of the*, volume 3, IEEE, pages. 1956–1961, 1995.

[37] KRISHNASWAMI, V. and G. RIZZONI, «A survey of observer based residual generation for FDI», *In Proc. IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages. 34–37, 1994.

[38] KRÖGER, T., *On-Line Trajectory Generation in Robotic Systems*, *Springer Tracts in Advanced Robotics*, volume 58, Springer, Berlin, Heidelberg, Germany, January 2010.

[39] LEUSCHEN, M. L., I. D. WALKER and J. R. CAVALLARO, «Fault residual generation via nonlinear analytical redundancy», *Control Systems Technology, IEEE Transactions on*, **13**(3), pages. 452–458, 2005.

[40] LÉVINE, J., *Analysis and control of nonlinear systems*, Springer Berlin, 2009.

[41] LÉVINE, J., «On necessary and sufficient conditions for differential flatness», *Applicable Algebra in Engineering, Communication and Computing*, **22**(1), pages. 47–90, 2011.

[42] LOOZE, D., J. WEISS, J. ETERNO and N. BARRETT, «An automatic redesign approach for restructurable control systems», *Control Systems Magazine, IEEE*, **5**(2), pages. 16–22, 1985.

[43] LÓPEZ FERNÁNDEZ, J. L. and C. RAVENTÓS OLIVELLA, «Model based fault detection using differential flatness», *5th AIAA-PEGASUS Student conference*, 2009.

[44] LOUEMBET, C., *Génération de trajectoires optimales pour systèmes différentiellement plats: application aux manoeuves d'attitude sur orbite*, PhD. thesis, 2007.

[45] LOUEMBET, C., F. CAZAURANG and A. ZOLGHADRI, «Motion planning for flat systems using positive B-splines: An LMI approach», *Automatica*, **46**(8), pages. 1305–1309, 2010.

[46] LOZANO-PEREZ, T., «A simple motion-planning algorithm for general robot manipulators», *Robotics and Automation, IEEE Journal of*, **3**(3), pages. 224–238, 1987.

[47] LUNZE, J. and J. RICHTER, «Control Reconfiguration: Survey of Methods and Open Problems», Bericht Nr. 2006.08, Ruhr-University of Bochum, June 2006.

[48] MAI, P., C. JOIN and J. REGER, «An example of flatness based fault tolerant control using algebraic derivative estimation», in *IAR Workshop, Nancy*, 2006.

[49] MAI, P., C. JOIN, J. REGER *et al.*, «Flatness-based Fault Tolerant Control of a nonlinear MIMO system using algebraic derivative estimation», in *3rd IFAC Symposium on System, Structure and Control, SSC'07*, 2007.

[50] MARTIN, P., *Contribution à l'étude des systèmes différentiellement plats*, PhD. thesis, 1992.

[51] MARTIN, P., S. DEVASIA and B. PADEN, «A different look at output tracking: Control of a VTOL aircraft», *Automatica*, **32**(1), pages. 101–107, 1996.

[52] MARTIN, P. and P. ROUCHON, «Any (controllable) driftless system with m inputs and m+ 2 states is flat», in *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, volume 3, IEEE, pages. 2886–2891, 1995.

[53] MARTÍNEZ TORRES, C., L. LAVIGNE, F. CAZAURANG, E. ALCORTA GARCIA and D. DIAZ, «Control reconfiguration for differentially flat systems.», in *Congreso Nacional de Control Automático (AMCA). Ensenada, Mexico*, 2013.

[54] MARTINEZ TORRES, C., L. LAVIGNE, F. CAZAURANG, E. ALCORTA GARCIA and D. DIAZ, «Fault detection and isolation on a three tank system using differential flatness.», in *European Control Conference. Zurich, Switzerland*, 2013.

[55] MARTÍNEZ TORRES, C., L. LAVIGNE, F. CAZAURANG, E. ALCORTA GARCIA and D. DIAZ, «Fault tolerant control of a three tank system: a flatness based approach», in *2nd International Conference on Control and Fault-Tolerant Systems. Nice, France*, 2013.

[56] MBOUP, M., C. JOIN and M. FLIESS, «A revised look at numerical differentiation with an application to nonlinear feedback control», in *Control & Automation, 2007. MED'07. Mediterranean Conference on*, IEEE, pages. 1–6, 2007.

[57] MILAM, M. B., R. FRANZ, J. E. HAUSER and R. M. MURRAY, «Receding horizon control of vectored thrust flight experiment», *IEE Proceedings-Control Theory and Applications*, **152**(3), pages. 340–348, 2005.

[58] MOGENS, B., K. MICHEL, L. JAN and S. MARCEL, *Diagnosis and Fault-tolerant control*, second edition, Springer, 2006.

[59] MORIO, V., *Contribution au développement d'une loi de guidage autonome par platitude. Application à une mission de rentrée atmosphérique.*, PhD. thesis, Université Sciences et Technologies-Bordeaux I, 2009.

[60] MORIO, V., A. FALCOZ and F. CAZAURANG, «On the design of a flatness-based guidance algorithm for the terminal area energy management of a winged-body vehicle», in *Proc. of the 17th IFAC Symposium on Automatic Control in Aerospace, IFAC, Toulouse, France*, 2007.

[61] NAN, Z., D. ANTOINE, D. ANDREI and M.-C. FELIXO, «A differential flatness approach for rotorcraft fault detection», in *Control Conference, 2008. CCC 2008. 27th Chinese*, IEEE, pages. 237–241, 2008.

[62] NGUANG, S. K., P. ZHANG and S. X. DING, «Parity relation based fault estimation for nonlinear systems: An LMI approach», *International Journal of Automation and Computing*, **4**(2), pages. 164–168, 2007.

[63] NONAMI, K., F. KENDOUL, S. SUZUKI, W. WANG and D. NAKAZAWA, *Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles*, Springer Publishing Company, Incorporated, 2010.

[64] NØRGAARD, M., N. K. POULSEN and O. RAVN, «New developments in state estimation for nonlinear systems», *Automatica*, **36**(11), pages. 1627–1638, 2000.

[65] NOURA, H., D. THEILLIOL, J.-C. PONSART and A. CHAMSEDDINE, *Fault-tolerant control systems: Design and practical applications*, Springer, 2009.

[66] RICHTER, J., *Reconfigurable Control of Nonlinear Dynamical Systems*, PhD. thesis, Fakultät für Elektrotechnik, Ruhr-Universität Bochum, 2010.

[67] RICHTER, J. H., *Reconfigurable control of nonlinear dynamical systems: a fault-hiding approach*, volume 408, Springer, 2011.

[68] ROTHFUSS, R., J. RUDOLPH and M. ZEITZ, «Flatness based control of a nonlinear chemical reactor model», *Automatica*, **32**(10), pages. 1433–1439, 1996.

[69] STAROSWIECKI, M., «Fault tolerant control: the pseudo-inverse method revisited», in *Proc. 16th IFAC World Congress, Prague, Czech Republic*, 2005.

[70] STENGEL, R. F., «Intelligent failure-tolerant control», *Control Systems, IEEE*, **11**(4), pages. 14–23, 1991.

[71] STUMPER, J.-F., F. SVARICEK and R. KENNEL, «Trajectory Tracking Control with Flat Inputs and a Dynamic Compensator», *arXiv preprint arXiv:1211.5759*, 2012.

[72] SURYAWAN, F., J. DE DONA and M. SERON, «Fault detection, isolation, and recovery using spline tools and differential flatness with application to a magnetic levitation system», in *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on*, IEEE, pages. 293–298, 2010.

[73] VAN NIEUWSTADT, M. J. and R. M. MURRAY, «Real-time trajectory generation for differentially flat systems», *International Journal of Robust and Nonlinear Control*, **8**(11), pages. 995–1020, 1998.

[74] VASILJEVIC, L. K. and H. K. KHALIL, «Error bounds in differentiation of noisy signals by high-gain observers», *Systems & Control Letters*, **57**(10), pages. 856–862, 2008.

[75] VENKATASUBRAMANIAN, V., R. RENGASWAMY and S. N. KAVURI, «A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies», *Computers & Chemical Engineering*, **27**(3), pages. 313–326, 2003.

[76] VENKATASUBRAMANIAN, V., R. RENGASWAMY, S. N. KAVURI and K. YIN, «A review of process fault detection and diagnosis: Part III: Process history based methods», *Computers & Chemical Engineering*, **27**(3), pages. 327–346, 2003.

[77] VENKATASUBRAMANIAN, V., R. RENGASWAMY, K. YIN and S. N. KAVURI, «A review of process fault detection and diagnosis: Part I: Quantitative model-based methods», *Computers & chemical engineering*, **27**(3), pages. 293–311, 2003.

[78] YAN, X.-G. and C. EDWARDS, «Nonlinear robust fault reconstruction and estimation using a sliding mode observer», *Automatica*, **43**(9), pages. 1605–1614, 2007.

[79] YAN, X.-G. and C. EDWARDS, «Robust sliding mode observer-based actuator fault detection and isolation for a class of nonlinear systems», *International Journal of Systems Science*, **39**(4), pages. 349–359, 2008.

[80] ZEITZ, M., «The extended Luenberger observer for nonlinear systems», *Systems & Control Letters*, **9**(2), pages. 149–156, 1987.

[81] ZERAR, M., F. CAZAURANG and A. ZOLGHADRI, «Robust tracking of nonlinear MIMO uncertain flat systems», in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 1, IEEE, pages. 536–541, 2004.

[82] ZHANG, N., A. DONCESCU, A. C. B. RAMOS and F. MORA-CAMINO, «Fault Detection for Difference Flat Systems», in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 2, 2012.

[83] ZHANG, Y. and J. JIANG, «Bibliographical review on reconfigurable fault-tolerant control systems», *Annual Reviews in Control*, **32**(2), pages. 229–252, 2008.

[84] ZHOU, K., J. C. DOYLE, K. GLOVER *et al.*, *Robust and optimal control*, volume 40, Prentice Hall New Jersey, 1996.

LIST OF FIGURES

LIST OF TABLES

APPENDIX A

TRAJECTORY GENERATION BY POLYNOMIAL APPROACH

This appendix recalls the construcion of trajectories, by using the polynomial approach. More advanced approaches, can be found in [libro trajectorias].

Let us define the initial and final conditions f_{ini} and f_{fin}, the trajectory generation problem consists in create a function $f(t)$ which fulfills those constraints.

This is a boundary condition problem, that can be easily solved by considering polynomial functions such as:

$$f(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + ... + a_n t^n \tag{A.1}$$

Where a_i when $i = 1, 2...n$ are polynomial coefficients, and t is the time.

The degree n of the polynomial depends on the number of boundary conditions that must be verified and on the desired “smoothness” of the trajectory. This degree has to be at least equals to the number of constraints, minus one.

Mathematically, these conditions may be expressed in matrix form as:

$$M * a = b \tag{A.2}$$

Where M is a known $(n+1)*(n+1)$ matrix, composed by time part of the equation A.1, b is the vector containing the known constraints. a contains the unknown coefficients.

The value of the coefficients can be easily computed by using the next expression

$$a = M^{-1}b \tag{A.3}$$

APPENDIX B

SMITH NORMAL FORM

This appendix is intended to present the definition of the Smith normal form, and give a recursive algorithm to obtain it. The Smith decomposition REF in a useful tool in mathematics which is specially helpful when working with principal ideal domain. Which is in fact used in many areas related to mathematics and engineering.

The unimodular [1] matrices of the principal ideal domain are invertible [2]. Besides each matrix M defined over the principal ideal domain admits a diagonal decomposition, known as Smith decomposition.

Let us define a principal ideal domain A, a polynomial matrix $M \in \mathcal{M}_{n,m}(A)$, it exists matrices $V \in \mathcal{U}_n(A)$ and $U \in \mathcal{U}_m(A)$, where $\mathcal{U}_n(A)$ and $\mathcal{U}_m(A)$ denotes the group of unimodular matrices of size $n * n$ and $m * m$ over A. Such as

$$VMU = \begin{cases} (\Delta \; 0_{n,m-n}) & if \; n < m \\ \begin{pmatrix} \Delta \\ \\ 0_{n-m,m} \end{pmatrix} & if \; n > m \end{cases} \tag{B.1}$$

[1]A square matrix which its determinant is equal to +1 or -1.

[2]A square matrix A is called invertible if there exists an n-by-n matrix B such that $AB = BA = I_n$

The right part of the equation B.1 is known as the Smith form of M, where $\Delta = diag\{\delta_1, ..., \delta_\sigma, 0, ..., 0\}$ is a diagonal matrix of size $n * n$ (resp. $m * m$), where the elements $(\delta_1, ...\delta_\sigma)$ are such that δ_i is a non-null polynomial for $i = 1, ...\sigma$ and it is a divider of σ_j for every $\sigma \geq j \geq i$.

Where V is a product of elementary row matrices, and U is a product of elementary column matrices.

Example B.1 *Compute the Smith decomposition of B.2*[3]

$$M = \begin{bmatrix} 1+x^2 & x \\ x & 1+x \end{bmatrix} \tag{B.2}$$

In order to compute the Smith decomposition of B.2, it is necessary to realize the operations in rows and columns as follows:

$$
\begin{array}{cccc|cc|cc}
 & & \multicolumn{2}{c}{Work} & & & \multicolumn{2}{c}{Work} \\
 & & \multicolumn{2}{c}{on\ rows} & & & \multicolumn{2}{c}{on\ columns} \\
 & & \multicolumn{2}{c}{\Downarrow} & & & \multicolumn{2}{c}{\Downarrow} \\
 & & 1 & 0 & 1+x^2 & x & 1 & 0 \\
 & & 0 & 1 & x & 1+x & 0 & 1 \\ \hline
R_1 \to R_1 - (x * R_2) & \Longrightarrow & 1 & -x & 1 & -x^2 & 1 & 0 \\
 & & 0 & 1 & x & 1+x & 0 & 1 \\ \hline
C_2 \to C2 - (x^2 * C_1) & \Longrightarrow & 1 & -x & 1 & 0 & 1 & x^2 \\
 & & 0 & 1 & x & 1+x+x^3 & 0 & 1 \\ \hline
R_2 \to R_2 - (x * R_1) & \Longrightarrow & 1 & -x & 1 & 0 & 1 & x^2 \\
 & & -x & 1+x^2 & 0 & 1+x+x^3 & 0 & 1 \\
 & & \multicolumn{2}{c}{\Uparrow} & \multicolumn{2}{c}{\Uparrow} & \multicolumn{2}{c}{\Uparrow} \\
 & & \multicolumn{2}{c}{V} & \multicolumn{2}{c}{M} & \multicolumn{2}{c}{U}
\end{array}
$$

[3] Example borrowed from www.numbertheory.org

APPENDIX C

2-NORM

The 2-Norm of a matrix A is defined as:

$$\|A\|_2 = \sqrt{\lambda_{max}} \tag{C.1}$$

Where λ_{max} is the largest eigenvalue such that $A^*A - \lambda I$[1] is singular[2].

When A is not singular the 2-Norm is defined as follows:

$$\|A^{-1}\|_2 = \frac{1}{\sqrt{\lambda_{min}}} \tag{C.2}$$

Where λ_{min} is the smallest eigenvalue such that $A^*A - \lambda I$ is singular.

[1] A^* is the conjugate transpose of A

[2] A matrix A is called singular if its determinant is zero, A singular matrix is not invertible