ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

ARCES - ADVANCED RESEARCH CENTER ON ELECTRONIC SYSTEMS FOR INFORMATION AND COMMUNICATION TECHNOLOGIES E. DE CASTRO

DOTTORATO DI RICERCA IN TECNOLOGIE DELL'INFORMAZIONE

CICLO XXVII

Settore Concorsuale di afferenza: 09/G1

Settore Scientifico disciplinare: ING-INF/04

FAULT DETECTION, DIAGNOSIS AND ACTIVE FAULT TOLERANT CONTROL FOR A SATELLITE ATTITUDE CONTROL SYSTEM

Tesi di dottorato presentata da

Pietro Baldi

Coordinatore di dottorato	Relatore
Prof. Claudio Fiegna	Prof. Paolo Castaldi

Esame finale anno 2015

Abstract

Modern control systems are becoming more and more complex and control algorithms more and more sophisticated. Consequently, Fault Detection and Diagnosis (FDD) and Fault Tolerant Control (FTC) have gained central importance over the past decades, due to the increasing requirements of availability, cost efficiency, reliability and operating safety.

This thesis deals with the FDD and FTC problems in a spacecraft Attitude Determination and Control System (ADCS). Firstly, the detailed nonlinear models of the spacecraft attitude dynamics and kinematics are described, along with the dynamic models of the actuators and main external disturbance sources. The considered ADCS is composed of an array of four redundant reaction wheels. A set of sensors provides satellite angular velocity, attitude and flywheel spin rate information. Then, general overviews of the Fault Detection and Isolation (FDI), Fault Estimation (FE) and Fault Tolerant Control (FTC) problems are presented, and the design and implementation of a novel diagnosis system is described. The system consists of a FDI module composed of properly organized model-based residual filters, exploiting the available input and output information for the detection and localization of an occurred fault. A proper fault mapping procedure and the nonlinear geometric approach are exploited to design residual filters explicitly decoupled from the external aerodynamic disturbance and sensitive to specific sets of faults. The subsequent use of suitable adaptive FE algorithms, based on the exploitation of radial basis function neural networks, allows to obtain accurate fault estimations.

Finally, this estimation is actively exploited in a FTC scheme to achieve a suitable fault accommodation and guarantee the desired control performances. A standard sliding mode controller is implemented for attitude stabilization and control. Several simulation results are given to highlight the performances of the overall designed system in case of different types of faults affecting the ADCS actuators and sensors.

Acknowledgments

Firstly, at the end of these three years, I wish to express my utmost gratitude and appreciation to my Ph.D. tutor, Prof. Paolo Castaldi, for his academic support and guidance, valuable suggestions and insightful criticism, and for his friendship, patience, constant spur and encouragement during these years.

Secondly, I wish to thanks Prof. Silvio Simani, from the Department of Engineering of the University of Ferrara, and Ing. Nicola Mimmo, from the Department of Electrical, Electronic and Information Engineering of the University of Bologna, for their friendship, helpful comments and contributions to the realization of this thesis.

Moreover, I would like to thank also all the other my colleagues at the laboratories of the Aerospace Engineering faculty in Forlì for their friendship and support during this experience, in particular Prof. Matteo Zanzi, Ing. Antonio Ghetti and Ing. Alessandro Mirri.

Part of this work has been developed while I was visiting the Department of Electrical Engineering of the Technical University of Denmark in Kogens-Lingby (DK). I want to express my sincere thanks to Prof. Mogens Blanke, who was my supervisor during my stay and gave me valuable suggestions. I wish also to thank, beside the staff of the Department, all the people of the Automation and Control Group who made my stay there pleasant and profitable.

Finally, a very special thanks to my parents and relatives for their unconditioned love and continuous support and encouragement in persevering in my research activity, and without whom this work would have been much harder to carry out.

Contents

List of Figures

List of Tables

1 INTRODUCTION

This chapter introduces the central themes this thesis deals with. First, the general aspects of fault diagnosis and fault tolerant control are introduced. Then, the main contributions and the organization of this thesis are outlined.

1.1 Fault Diagnosis and Fault Tolerant Control

Modern control systems are becoming more and more complex and control algorithms more and more sophisticated. Consequently, the requirements of availability, cost efficiency, reliability, operating safety and environmental protection are of major importance, not only for safety-critical systems, but also for many other advanced systems. For safety-critical systems, the consequences of faults can be extremely serious in terms of human mortality, environmental impact and economic loss. Therefore, there is a growing need for online supervision and fault diagnosis to increase the reliability of such safety-critical systems. Early indications concerning which faults are developing can help to avoid system breakdown, mission abortion and catastrophes. For systems which are not safety-critical, online fault diagnosis techniques can be used to improve system and cost efficiency, maintainability, availability and reliability.

Since the early days of the development of automatic supervisory control, very significant attention to these safety and security issues were paid by academic researchers (Clark *et al.* 1975; Patton *et al.* 1989). This is especially the case in aeronautics and astronautics, which have stringent requirements on stability, performance and reliability. Heavy demands can be placed on an automatic system to help to avoid repetition of tragedies and/or critical economic loss.

The causes leading to undesired system behaviours or unusual sensing behaviours are called *faults*, which are conceptually defined as unpermitted deviations of at least one characteristic property or parameter of the system from the acceptable/usual/standard operating conditions, as according to Isermann and Ballé (1997). As a consequence of each fault occurring during operations, *failure* is defined as the complete breakdown of a system component or function. It describes the situation that the system no longer performs the required function (Isermann and Ballé 1997). Faults occurring in actuators, sensors or other system components may lead to unsatisfactory performance or, even worse, instability. The monitoring system takes the responsibility of detecting and diagnosing unanticipated behaviours. This supervisory system is the so-called *Fault Detection Isolation/Diagnosis* (FDI/FDD) system. FDI/FDD systems should not only provide alarms when a malfunction occurs in the supervised system, but should also provide a classification and identification of the erroneous behaviour occurring during the entire system operation. Moreover, the FDI/FDD systems should provide information and reports to prevent further loss of system function. Generally speaking, FDD goes slightly further than FDI by including the possibility of estimating the effect of the fault and/or diagnosing the effect or severity of the fault. Hence, the term FDD also covers the capability of isolating or locating a fault. Subsequently, the operators/automatic control systems should be informed of the fault situations and proper actions should be taken to avoid total system breakdown and catastrophe (Patton *et al.* 1989). Hence, a reliable and affordable fault diagnosis system is very critical from safety and sustainability perspective and plays a significantly important role in many applications (Chen and Patton 1999).

The traditional fault diagnosis approach localises the faults by making use of *hardware redundancy* (all system components are replicated, including actuators, sensors, computers to measure and/or control a particular variable). The location of a fault can be inferred using a majority voting scheme, where three or more redundant lanes of system hardware are used to provide the same function. However, the cost, complexity and volume of many modern system devices make the hardware redundancy approach much less applicable in terms of maintenance and operational costs, weight restriction or even in terms of strictly regulated ecological requirements.

An alternative to the use of redundant hardware is to develop systems that have *analytical redundancy* or *functional redundancy* based on the use of *model-based* information. Analytical redundancy effectively transforms the hardware redundancy into realisable software estimation problems. Redundant or additional/repeated estimates of the measured signals are used to derive estimates of other variables of the system without the use of additional measurement sensors (Chen and Patton 1999). The only required information for the model-based FDI/FDD approach is the availability of a valid system model and the use of the measured inputs and outputs of the system being monitored. However, in order to achieve reliability and robustness, special methods must be used to ensure that the estimated variables are faithful replicas of the measured quantities. The expected outcomes from the model-based FDI/FDD approach are multiple symptoms (residuals or fault estimation signals) indicating the differences between nominal and faulty system status in a timely manner.

Therefore, the increasing demands for system safety, reliability, maintainability and survivability in many application fields, including aeronautics and aerospace, has motivated and accelerated the development of different types of FDI/FDD approaches, and in particular several model-based strategies (Gertler 1998; Chen and Patton 1999; Blanke *et al.* 2006; Simani *et al.* 2003; Edwards *et al.* 2010; Isermann 2005, 2011; Ding 2013). In particular, the exploitation of model-based approaches is often strongly necessary for aeronautical and aerospace applications, in particular due to lack of space, cost and weight limitations in small aircrafts and spacecrafts, which make hard to implement multiple hardware redundancies.

Additionally, these increasing demands from the supervised system for safety, reliability, availability, maintainability, survivability and sustainability motives also to develop *Fault Tolerant Control* (FTC) schemes with the capability of tolerating system malfunctions preventing loss of life, mitigating against hazards, and avoiding economic loss, etc. FTC is also expected to maintain desirable and robust performance and stability properties in the case of malfunctions in actuators, sensors or other system components (Patton 1997a, 1997b; Blanke *et al.* 2006).

As a consequence, FDI/FDD information are certainly important in FTC, *e.g.* when the control system is reconfigured only subsequently to the detection of an occurred fault. If the location, fault onset time and severity of the fault are determined, from either an FDI residual signal or using estimates of a fault, then appropriate action can be taken to switch or reconfigure the control system either using on-line or off-line computed control laws corresponding to various potential fault scenarios. When the FTC system makes use of fault information for reconfiguration this is known as *Active FTC* (AFTC), whilst the alternative *Passive FTC* (PFTC) methods do not require fault diagnosis information and are thus based mainly on robust control ideas (Patton 1997a, 1997b).

AFTC schemes are used to trigger specific control actions in real-time (based on fault information) to prevent plant damage as a consequence of malfunctions and ensure system availability and sustainability based on the use of redundancy (in either analytical or hardware forms). AFTC can also be used to ensure that the control system performance is not degraded when there is a loss of efficiency in closed-loop system components, *i.e.* corresponding to minor or incipient fault conditions (Patton 1997a, 1997b).

Hence, in an AFTC mechanism, sufficient real-time fault information is required to accommodate to the effects of faults by a reconfiguration mechanism. The AFTC performance is strongly affected by the degree to which accurate fault information is available. Whilst residual-based FDI methods can provide a high degree of fault information accuracy, a preferable approach is to use on-line Fault Estimation (FE) signals that are designed to robustly reconstruct the time-variation of each occurred fault. The reconfiguration scheme in this case may use the FE signal to compensate for the fault in the closed-loop system. The more precisely the fault information is provided by on-line FE, the more successfully the AFTC system performs (Patton 1997a, 1997b; Zhang and Jiang 2008).

1.2 Thesis Contributions

In the context of aerospace applications, this thesis deals with the FDI, FDD and FTC problems for a spacecraft attitude control application, in particular focusing on fault tolerant control and diagnosis of possible faults affecting the actuators and sensors of an Attitude Determination and Control System (ADCS) of a Low Earth Orbit (LEO) satellite. This dissertation aims to summarize and extend the results previously shown in (Baldi *et al.* 2010a, 2010b, 2012, 2013, 2014b, 2015). Differently from (Baldi *et al.* 2010a, 2010b, 2012, 2013, 2014b), this thesis extends the problem of fault diagnosis also to sensor faults, in addition to actuator faults only.

A novel fault diagnosis and fault tolerant control scheme is developed for the detection, isolation, estimation and accommodation of possible faults affecting the control torques provided by reaction wheel actuators, flywheel spin rate measurements, attitude and angular velocity measurements provided by the sensors of the satellite ADCS. The proposed diagnosis scheme is based on the exploitation of a Fault Detection and Diagnosis (FDD) system which is composed of a Fault detection and Isolation (FDI) module and a Fault Estimation (FE) module.

For simplicity, the overall ADCS can be considered to be composed by two parts: the Attitude Control System (ACS), consisting of the actuators and their dedicated flywheel spin rate sensors, and the Attitude Determination System (ADS), consisting of the satellite attitude and angular velocity sensors.

In particular, the satellite ACS considered in this thesis is assumed to be composed of an array of four (redundant) reaction wheel actuators generating reaction torques for attitude control on the three axes of the satellite, and embedding also the actuator sensors measuring the flywheel spin rates. The four reaction wheels are arranged in a tetrahedral configuration. In practice, this configuration allows having an overall three-axis zero momentum bias even if the wheels run with a momentum bias. Moreover, the presence of a fourth redundant actuator allows to maintain the complete attitude controllability also in case of failure of one actuator. The overall ADCS is completed by the ADS, which is assumed to be composed of two star sensors and three rate gyro sensors for the determination of the spacecraft attitude and angular velocity, respectively.

The designed FDI module is composed of a set of scalar model-based residual generators, which are organized in four independent banks of filters working in parallel, each of them specifically dedicated to the detection and isolation of possible faults affecting specific sets of components of the ADCS. The first two banks are specifically exploited for the detection and isolation of faults occurred in the actuators and their actuator sensors, *i.e.* in the actuator subsystems of the ACS. The other two banks are specifically exploited for the detection and isolation of faults occurred in the attitude and angular velocity sensors of the spacecraft, *i.e.* in the ADS.

Moreover, considering the first couple of banks of residual filters for the detection and isolation of faults in the actuators and their actuator sensors, the first bank, called Residual Bank n.1 (RB1), will be referred from now on also as *local*. This is done because the designed local residual filters are based on the dynamic equations of the actuators, and thus rely only on local measurements of the spin rates of the actuator flywheels. Now, it is worth observing that the internal electrical models of actuators and sensors have been neglected in this thesis, *i.e.* no fault diagnosis is performed using local electrical measurements of current or voltage or other types of direct internal checks in the supervised system components. Only sensor measurements, which are intended and exploited for the overall attitude control purpose, are assumed to be available to the diagnosis system. The second bank of residual filters, called Residual Bank n.2 (RB2), will be referred from now on also as *global*. This is done because the designed global residual filters are based on both the dynamic and kinematic equations of the spacecraft model and the dynamic equations of the actuators, and thus rely also on global measurements of the spacecraft attitude and angular velocity in addition to the local flywheel spin rate measurements. These global residual filters result to be explicitly decoupled from the aerodynamic disturbance thanks to the application of the NonLinear Geometric Approach (NLGA) formally developed by De Persis and Isidori (2000, 2001). Due to the aerodynamic disturbance uncertainty in Low Earth Orbit satellites, this disturbance decoupling allows to obtain better diagnosis performances. In fact, it is unnecessary to take account of the uncertainty of knowledge of the aerodynamic disturbance parameters in the selection of the residual thresholds, allowing the detection of smaller faults, and hence subsequently achieving better control performances. On the other hand, the design procedure of the local residual filters do not need to exploit the NLGA for disturbance decoupling since the exploited actuator dynamic equations, do not include exogenous disturbance uncertainties to be decoupled. The application of the NLGA is immediate and straightforward.

The general procedure proposed by Mattone and De Luca (2006b) is used for modelling of the spin rate sensor faults in order to obtain a new spacecraft nonlinear dynamic model, which is affine with respect to both the actuator and sensor faults and suitable to the application of the NLGA.

Considering the second couple of banks, called Residual Bank n.3 (RB3) and Residual Bank n.4 (RB4) respectively, they are specifically exploited to detect and isolate faults occurred in the ADS, *i.e.* in the satellite attitude and angular velocity sensors. The design procedure of these residual filters again exploits the NLGA in order to obtain scalar residual filters sensitive only to possible faults affecting specific sets of sensors for fault isolation purpose. However, also in this case, the design procedure do not need to exploit the NLGA for disturbance decoupling since the exploited kinematic equations of the spacecraft model do not include exogenous disturbance uncertainties to be decoupled. The designed filters in these two banks are based on the same mathematical model equations, but each of them is fed by the same set of angular velocity measurements and a different attitude quaternion measurement vector provided by one of the two attitude sensors included in the ADS. The exploitation of a double redundancy of the attitude sensors is required since, in general, it would not be possible to distinguish between faults on an angular velocity sensor or attitude sensor with only a single attitude measurement available.

The overall detection and isolation procedure for the considered actuator and sensor faults is carried out by exploiting a cross-check of all the generated residuals, by means of a proper decision logic and residual comparison scheme. In this context, the joint use the four banks of residual filters in the FDI system allows riding over the limitations of each bank of residual filters for the detection and isolation of faults in the supervised subsystems.

In fact, for example, a FDI module effectively exploiting only the designed local filters of RB1 is not sufficient in order to guarantee the accurate detection and isolation of faults affecting the flywheel spin rate sensors or attitude control actuators.

A FDI module exploiting only these local residual filters would be not able to discern if an occurred fault is affecting the actuated control torque or the corresponding flywheel spin rate sensor of a specific actuator subsystem (*i.e.* the system composed by the actuator itself and its corresponding flywheel spin rate sensor). In fact, the designed local residual filters can only isolate the generic faulty actuator subsystem, without specify which type of fault has occurred. The identification of the fault type can be achieved only by exploiting also the global filters of RB2.

The overall FDD system is completed by a Fault Estimation (FE) module, which consists of a bank of adaptive observers exploited to obtain accurate and quick fault estimates. This bank of adaptive observers is based on a Radial Basis Function Neural Network (RBF-NN) (Buhmann 2003; Wang *et al.* 2011; Baldi *et al.* 2013, 2014a, 2015; Castaldi *et al.* 2014). The on-line learning capability of the Radial Basis Function Neural Network allows obtaining accurate adaptive estimates of the occurred faults. Moreover, the use of a Radial Basis Function Neural Network allows designing generalized fault estimation adaptive observers which do not need any a priori information about the fault internal model. The outputs of the fault estimation module (*i.e.* the fault estimates) are enabled once a fault is correctly detected and isolated by the previously designed FDI module. However, it is worth noting that all the fault estimation adaptive observers are active from the beginning of the simulation, and only the output of a specific estimation filter is enabled once a fault has been detected and correctly isolated, with the assumption of a single fault occurring at a time. In this way, no inconsistent fault estimates are provided by the FDD system.

Finally, an Active Fault Tolerant Control (AFTC) system is realized by implementing a fault tolerant strategy, based on the information from the FDI/FDD system.

In case of actuator faults, a fault accommodation scheme is exploited for soft faults, when the faulty actuator is yet operative but with a degraded performance, by exploiting the fault estimation information from the FE module. On the contrary, a control reconfiguration scheme is used in case of hard actuator faults (*i.e.* total actuator failures), by excluding the effects of the faulted actuator in the system by exploiting the actuator redundancy and using the control inputs of the other actuators appropriately to maintain the complete attitude controllability. In this way, a Fault Detection, Isolation and Recovery (FDIR) scheme is realized, without directly exploiting any fault estimation information from the FE module. In case of sensor faults, a fault accommodation scheme is used by directly exploiting the fault estimation information from the FE module.

The performances of the proposed fault diagnosis and fault tolerant control strategies have been evaluated when applied to a detailed nonlinear spacecraft attitude model taking account also of measurement noise, and exogenous disturbance signals. In particular, these exogenous disturbance terms are represented by aerodynamic and gravitational disturbances. However, as the gravitational disturbance model is almost perfectly known, the FDI robustness is achieved by exploiting an explicit disturbance decoupling method, based on the NLGA, applied only to the aerodynamic force term. This term represents the main source of uncertainty in the satellite dynamic model, mainly due to the lack of knowledge of the accurate values of air density and satellite drag coefficient.

The quick and accurate detection, isolation and estimation of faults affecting both the attitude control actuators, the flywheel spin rate sensors and attitude and angular velocity sensors of the spacecraft results to be a key point in order to guarantee the desired attitude control performances through the subsequent fault accommodation in the proposed AFTC schemes. Several simulation results are given for different fault scenarios for both the actuators and sensors. The obtained results highlight that the proposed diagnosis scheme can deal with the most significant, generic types of faults.

1.3 Thesis Outline

The thesis is organised as follows. **Chapter 2** provides a description of the nonlinear dynamic and kinematic models of a rigid spacecraft and of the considered external disturbance models. A description of the main reference frames and notations for attitude representation is also provided, and the actuators and sensors implemented in the considered spacecraft ADCS are described.

Chapter 3 provides an overview of the FDI problem, together with the general description of the residual generation method and fault detection and isolation scheme. The sensor and actuator fault modelling method and NLGA for affine nonlinear systems, which are exploited for disturbance decoupling and fault detection, isolation and estimation, are illustrated.

Chapter 4 illustrates the design and practical implementation of a complete FDI system for the spacecraft ADCS, consisting in four banks of residual generators decoupled from the external aerodynamic disturbance torque acting on the satellite, designed by exploiting the NLGA and the used fault modelling method. Moreover, the fault detection and isolation procedure, based on the cross-check of the provided diagnostic signals, is described along with the corresponding decision logic. Simulation results for FDI are given in case of sensor and actuator fault occurrence. It is highlighted how different types of faults can be accurately and promptly detected and isolated.

Chapter 5 provides an overview of the FDD problem, together with a general description of the design method of the adaptive fault estimation filters for sensor and actuator faults. A description of the RBF-NN exploited by the estimation filters is also provided, along with the corresponding adaptive laws of the neural network output layer weights.

Chapter 6 illustrates the design and practical implementation of the fault estimation filters of a complete FDD system for the considered spacecraft ADCS. The design of adaptive fault estimation filters based on the NLGA and RBF-NN and decoupled from the external aerodynamic disturbance torque is illustrated. Simulation results for FDD are given in case of sensor and actuator fault occurrence. It is highlighted how different types of faults can be accurately and promptly estimated.

Chapter 7 provides an overview of the FTC problem, together with a general description of the different fault tolerant control approaches exploited in this thesis in case of fault occurrence.

Chapter 8 illustrates the practical implementation of different FTC schemes in the considered spacecraft ADCS. The implementation of a SMC is illustrated. Although this control method results to be in general robust to model uncertainties, external disturbances and even faults on the control inputs, essentially consisting in a PFTC in these cases, the exploitation of AFTC approaches allows to obtain better performances during transient phases and in case of sensor faults. Simulation results for FTC are given in case of sensor and actuator fault occurrence, in order to highlight the system capability to recover from faults and/or enhance its performances. Finally, concluding remarks are drawn in **Chapter 9**.

2 SPACECRAFT ATTITUDE DYNAMICS

In general, the way that a spacecraft moves is described using orbit dynamics, which determines the position of the body along the orbital path, as well as attitude dynamics, which determines its orientation. In particular, since this thesis deals with the diagnosis of faults affecting the Attitude Determination and Control System (ADCS) of a spacecraft, this chapter focuses exclusively on the equations describing the attitude dynamics and kinematics of a rigid body. Attitude dynamics describes the orientation of a body in an orbit and can be explained using rotations. In addition, frames of reference and attitude rotations are discussed, along with environmental disturbance torques. This chapter concludes with information on the attitude determination and control system considered in the performed simulations.

2.1 Reference Frames

When examining attitude dynamics, it is important to describe the reference frames being used to give a basis for the rotations. Three main reference frames, or coordinate systems, are used in this thesis to describe the orientation, or attitude, of a spacecraft in orbit. These are F_i, F_o and F_b, the inertial, orbital, and body reference frames respectively (Wertz 1978; Wertz and Larson 1999). The formal definitions of these coordinate systems are given in the following subsections.

2.1.1 Inertial Reference Frame F_i

The inertial reference frame is a fixed non-spinning frame that is commonly used for attitude applications. Usually, the x_i direction points from the focus of the orbit to the vernal equinox ♈, the z_i direction is in the orbital angular velocity direction, and y_i is perpendicular to x_i and z_i. Vernal equinox is the point where the ecliptic crosses the Earth equator going from South to North on the first day of spring (*i.e.* the line from Earth's origin through the Sun on the first day of spring). Since a circular and equatorial Low Earth Orbit (LEO) is considered in this thesis, an Earth-Centred Inertial (ECI) coordinate system is used, in particular the geocentric equatorial coordinate system, which is a right-handed orthogonal coordinate system with origin located in the centre of the Earth, as shown in Fig. 2.1. In this reference frame, the z_i axis points through the geographic North Pole, or the axis of rotation of the Earth. The x_i axis is in the direction of the vernal equinox ♈, and the y_i direction is right-handed orthogonal. The Earth rotates with respect to the ECI coordinate system. It is common to use the geocentric equatorial coordinate system as the inertial reference frame for spacecraft control. The inertial frame is used as the reference to measure the attitude of the satellite with respect to the fixed stars.

x_i -axis pointing in the vernal equinox direction;

z_i -axis pointing upwards from the origin through the geographical North Pole;

y_i -axis completing the right hand system.

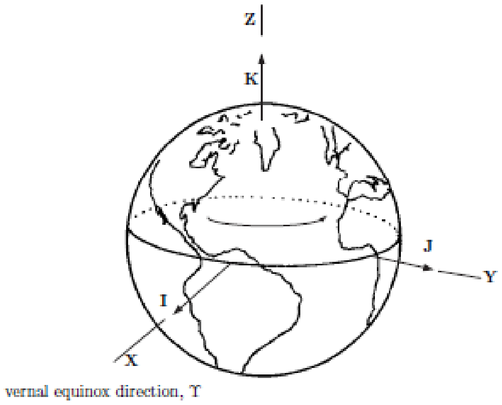


Figure 2.1 - Earth-centred inertial reference frame.

2.1.2 Orbital Reference Frame F_o

The orbital reference frame is a right-handed orthogonal coordinate system with origin located at the centre of mass of the spacecraft, and the motion of the frame depends on the current position of the satellite in orbit. In this thesis, the satellite is assumed to move along an equatorial and circular orbit. The orbital reference frame is also referred to as Local Vertical Local Horizontal (LVLH) frame. This reference frame is non-inertial because of orbital acceleration and the rotation of the frame with respect to the ECI reference frame. The z_o axis of the orbital frame is in the direction from the spacecraft to the Earth centre (nadir pointing), y_o is the direction opposite to the normal of the orbital plane, and x_o is perpendicular to y_o and z_o. Fig. 2.2 shows how the LVLH orbital frame is generally defined. In circular orbits, x_o corresponds to the direction of the spacecraft velocity along the orbital path. The three directions x_o, y_o, and z_o are also known as the roll, pitch, and yaw axes, respectively. The local orbital frame is used as the instantaneous reference to express the attitude of the satellite with respect to the Earth.

x_o-axis pointing in the direction of motion, tangential to the orbit;
z_o-axis pointing to the centre of the Earth (nadir;
y_o-axis completing the right-hand system.

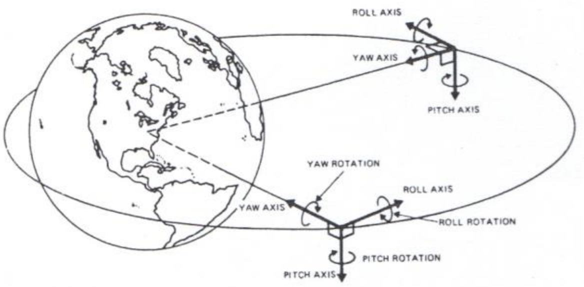


Figure 2.2 - Local Vertical Local Horizontal (LVLH) orbital reference frame.

2.1.3 Body Reference Frame F_b

Similarly to the orbital reference frame, the body reference frame has its origin at the centre of mass of the spacecraft. This right-handed orthogonal frame is fixed in the rotating body, and therefore is non-inertial. The relative orientation between the inertial or orbital frame and the body frame is the basis of attitude dynamics and control. The body reference frame is assumed to be aligned with the orbital reference frame in the nadir pointing condition of the satellite. Measurements taken by instruments on-board the satellite are often taken with respect to the body frame. Fig. 2.3 shows a comparison of the inertial, orbital and body frames.

x_b -axis pointing from the back to the front;
z_b -axis pointing from the top to the bottom;
y_b -axis completing the right hand system.

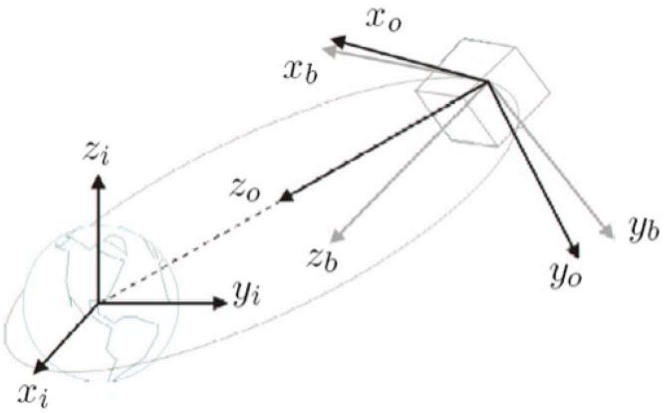


Figure 2.3 - Earth centred inertial, orbital and body reference frames.

2.1.4 Principal Axis of Inertia

Principal axes are a specific case of body-fixed reference frame. This right-handed orthogonal coordinate system has its origin at the centre of mass of the spacecraft, and it is oriented such that the moment of inertia tensor of the spacecraft is diagonal, whose nonzero elements are known as the principal moments of inertia. In particular, this is the body-fixed coordinate system actually exploited in this thesis for the description of the spacecraft attitude dynamics.

2.2 Attitude Representations

In aerospace applications, the term attitude refers to the orientation of a body-fixed orthogonal coordinate system with respect to another reference frame. The rotation matrix, Euler angles and quaternions are three different ways to represent the spacecraft attitude (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008).
It is worth noting that the attitude can be described mathematically with a minimum of three independent parameters for a rigid body. However, the usage of only three parameters may result in singularities. Hence, depending on the applications, parameterizations with more than three elements are also considered. However, even if the number of parameters are more than three, the number of independent parameters are still three in all the representations, others being related through constraint equation. The following Tab. 2.1 offers a comparison of Euler angles, quaternions and rotation matrix, which are the primarily used attitude representations, along with the their main advantages and disadvantages. Other representations exist, apart from the ones listed below, such as Euler axis-angle, Gibbs vector and Rodriguez parameters.

	Notation	Advantages	Disadvantages
Direction Cosine Matrix (DCM)	R	No singularities, ease of considering successive rotations	6 redundant parameters
Quaternion	$\bar{\mathbf{q}}=[q_1,q_2,q_3,q_4]^T$	No singularities, ease of considering successive rotations	1 redundant parameter
Euler angles	$\Theta=[\phi,\theta,\psi]^T$	No redundant parameters, physical interpretation	Singularities, trigonometric functions

Table 2.1 - Comparison of attitude representations.

The three attitude representations listed above are related to each other. Proper equations relating the three different parameterizations and detailed comparisons can be found in Kaplan (1976), Wertz (1978), Hughes (1986), Sidi (1997), Egeland and Gravdahl (2002), Tewari (2007) and Wie (2008).

2.2.1 Rotation Matrix

The rotation matrix, also referred to as Direction Cosine Matrix (DCM), is a non-minimal description of the rigid body's orientation with only three degrees of freedom. The rotation matrix can be interpreted in three different ways: as a coordinate transformation matrix mapping a vector represented in one coordinate frame to another frame, as a rotation of a vector within the same frame and finally as a description of the mutual orientation between two frames. Hence, the attitude of a rigid body with respect to a reference frame can be described by a matrix $\mathbf{R}$ representing the relative orientation between the two frames (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008).

The generic direction cosine matrix $\mathbf{R}_a^b$ is a transformation matrix which is composed of the direction cosine values between the initial coordinate system F_a and the target coordinate system F_b, and describes the rotation from F_a to F_b. Considering the base vectors of F_a given by the unit vectors $\vec{x}_a$, $\vec{y}_a$, $\vec{z}_a$ and the base vectors of F_b given by the unit vectors $\vec{x}_b$, $\vec{y}_b$, $\vec{z}_b$, the direction cosine matrix is defined as

$$\mathbf{R}_a^b=\begin{bmatrix}\cos\sphericalangle(\vec{x}_a,\vec{x}_b) & \cos\sphericalangle(\vec{y}_a,\vec{x}_b) & \cos\sphericalangle(\vec{z}_a,\vec{x}_b)\\ \cos\sphericalangle(\vec{x}_a,\vec{y}_b) & \cos\sphericalangle(\vec{y}_a,\vec{y}_b) & \cos\sphericalangle(\vec{z}_a,\vec{y}_b)\\ \cos\sphericalangle(\vec{x}_a,\vec{z}_b) & \cos\sphericalangle(\vec{y}_a,\vec{z}_b) & \cos\sphericalangle(\vec{z}_a,\vec{z}_b)\end{bmatrix}=\begin{bmatrix}\vec{x}_a\cdot\vec{x}_b & \vec{y}_a\cdot\vec{x}_b & \vec{z}_a\cdot\vec{x}_b\\ \vec{x}_a\cdot\vec{y}_b & \vec{y}_a\cdot\vec{y}_b & \vec{z}_a\cdot\vec{y}_b\\ \vec{x}_a\cdot\vec{z}_b & \vec{y}_a\cdot\vec{z}_b & \vec{z}_a\cdot\vec{z}_b\end{bmatrix}$$

This rotation matrix is an element in a special orthogonal group of order three $SO(3)$:

$$\mathbf{R}\in SO(3),\quad SO(3)=\left\{\mathbf{R}\,|\,\mathbf{R}\in\mathbb{R}^{3\times3},\mathbf{R}\text{ is orthogonal and }\det(\mathbf{R})=1\right\}$$

This means that rotation matrices have some useful properties:

$$\mathbf{R}=\mathbf{R}^{-1}=\mathbf{R}^T$$
$$\mathbf{R}\mathbf{R}^T=\mathbf{R}^T\mathbf{R}=\mathbf{I}_3$$
$$\det(\mathbf{R})=1$$

Since the dynamic model of the spacecraft incorporates several different coordinate frames, a way to convert a vector from one frame to another is needed. Exploiting the rotation matrix, the relationships between vectors expressed in different reference frames are given by the coordinate transformations

$$\mathbf{v}_b = \mathbf{R}_a^b \mathbf{v}_a$$
$$\mathbf{v}_a = \mathbf{R}_b^a \mathbf{v}_b = (\mathbf{R}_a^b)^T \mathbf{v}_b$$

where $\mathbf{R}_a^b$ is the rotation matrix describing the relative orientation of the generic frame F_a with respect to the reference frame F_b, $\mathbf{R}_b^a = (\mathbf{R}_a^b)^T$ is the corresponding inverse transformation, $\mathbf{v}_a$ is a vector expressed in F_a, and $\mathbf{v}_b$ is the same vector expressed in F_b.

When creating the rotation matrix using Euler angles, described in the next subsection, it is possible to combine principal rotations. Each of these principal rotations represents a transformation of a coordinate system about one of its basis vectors with a generic rotation angle. These transformations are described by the following three elementary rotation matrices:

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{2.1}$$

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{2.2}$$

$$\mathbf{R}_z(\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

where the subscriptions *x*, *y* and *z* denotes the axis of the coordinate system the rotation angles ϕ, θ and ψ revolves about, respectively.

A generic coordinate transformation matrix can be built from the composition of two or more consecutive rotations. In case of three consecutive rotations, for example, it can be defined as

$$\mathbf{R}_a^d = \mathbf{R}_c^d \mathbf{R}_b^c \mathbf{R}_a^b \tag{2.4}$$

where the matrices $\mathbf{R}_a^b$, $\mathbf{R}_b^c$ and $\mathbf{R}_c^d$ represent three consecutive rotations and $\mathbf{R}_a^d$ is the overall coordinate transformation matrix to pass from the generic coordinate system F_a to the coordinate system F_d. It is worth noting that the coordinate transformation matrix corresponding to a set of consecutive rotations about moving axes is obtained by pre-multiplying the different single rotation matrices, thus multiplying the elementary rotation matrices in Eq. (2.4) in the opposite order in which the rotations take place.

Two notations commonly used to describe these coordinate transformations and parameterize the corresponding rotation matrices are the Euler angles and unit quaternion, as it will be shown in the following.

2.2.2 Euler Angles

The Euler angles $\Theta=\begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$, also referred to as the roll, pitch and yaw angles respectively, are commonly used to describe the motion of rigid bodies that moves freely, like spacecrafts (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008).

As already stated, the spacecraft attitude can be described by the rotation between the body-fixed reference frame and another reference frame. Since in this thesis a LEO Earth-pointing satellite is considered, it is convenient to define the satellite attitude as the rotation between the body-fixed reference frame and the orbital reference frame. On the contrary, for a satellite pointing a generic star of sky portion, it could be more convenient to define the satellite attitude with respect to the inertial frame.

The rotation from the orbital reference frame to the body-fixed reference frame can be considered as a composite rotation, consisting for example of the following three elementary rotations, corresponding to Eqs. (2.1), (2.2) and (2.3), in the indicated order:

1. a rotation about the x_o-axis with roll angle ϕ (Eq. (2.1));
2. a rotation about the current (rotated) y-axis with pitch angle θ (Eq. (2.2));
3. a rotation about the current (rotated) z-axis with yaw angle ψ (Eq. (2.3)).

This sequence is called X-Y-Z rotation axis sequence. Twelve different orders and combinations of the elementary rotations of Eqs. (2.1), (2.2) and (2.3) can be considered to represent the attitude of a spacecraft. The resulting orthogonal transformation matrix for the rotation sequence considered above becomes

$$\mathbf{R}_o^b=\mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\phi)=\begin{bmatrix}i_o^b & j_o^b & k_o^b\end{bmatrix} \tag{2.5}$$

with $\mathbf{R}_x(\phi)$, $\mathbf{R}_y(\theta)$ and $\mathbf{R}_z(\psi)$ given by Eqs. (2.1), (2.2) and (2.3) respectively, which yields

$$\mathbf{R}_o^b(\Theta)=\begin{bmatrix}\cos\psi\cos\theta & \sin\psi\cos\phi+\cos\psi\sin\theta\sin\phi & \sin\psi\sin\phi-\cos\psi\sin\theta\cos\phi\\ -\sin\psi\cos\theta & \cos\psi\cos\phi-\sin\psi\sin\theta\sin\phi & \cos\psi\sin\phi+\sin\psi\sin\theta\cos\phi\\ \sin\theta & -\cos\theta\sin\phi & \cos\theta\cos\phi\end{bmatrix} \tag{2.6}$$

This matrix is also used for vector transformations from the orbital reference frame to the body-fixed reference frame. Considering the inverse rotation from the body-fixed reference frame to the orbital reference frame, the resulting coordinate transformation matrix becomes

$$\mathbf{R}_b^o=(\mathbf{R}_o^b)^{-1}=(\mathbf{R}_o^b)^T=\mathbf{R}_x^T(\phi)\mathbf{R}_y^T(\theta)\mathbf{R}_z^T(\psi) \tag{2.7}$$

which yields

$$\mathbf{R}_b^o(\Theta)=\begin{bmatrix}\cos\psi\cos\theta & -\sin\psi\cos\theta & \sin\theta\\ \sin\psi\cos\phi+\cos\psi\sin\theta\sin\phi & \cos\psi\cos\phi-\sin\psi\sin\theta\sin\phi & -\cos\theta\sin\phi\\ \sin\psi\sin\phi-\cos\psi\sin\theta\cos\phi & \cos\psi\sin\phi+\sin\psi\sin\theta\cos\phi & \cos\theta\cos\phi\end{bmatrix} \tag{2.8}$$

In the same way, two coordinate transformation matrices $\mathbf{R}_i^b(\Theta)$ and $\mathbf{R}_b^i(\Theta)$ can be defined, by considering the body-fixed and inertial coordinate systems as reference frames. In the following, to avoid misunderstandings, the subscripted Euler angles $\Theta_o=\begin{bmatrix}\phi_o & \theta_o & \psi_o\end{bmatrix}^T$ will represent the spacecraft attitude with respect to the orbital reference frame, whereas the subscripted Euler angles $\Theta_i = \begin{bmatrix} \phi_i & \theta_i & \psi_i \end{bmatrix}^T$ will represent the spacecraft attitude with respect to the inertial reference frame. The Euler angles represent an intuitive representation for the attitude of an object in a three-dimensional space. However, using Euler angles to describe the attitude kinematic may result in singularities for $\theta = \pm\frac{\pi}{2}$, corresponding to the rotation matrix in Eq. (2.6) being singular (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008). This singularity can be also shifted to different angles by changing the combination of elementary rotations, but the problem is always present in the mathematical description of the attitude and kinematics by using Euler angles. However, alternative notations, like the quaternions, can be used to avoid singularities. In fact, introducing a fourth parameter to represent the attitude can solve the problem. In this thesis, Euler angles will be used in the simulations only to best visualize the behaviour of the satellite and performance of the designed systems, due to their physical meanings.

2.2.3 Unit Quaternion

An alternative way to parameterize $SO(3)$ and describe the spacecraft attitude is through the use of quaternions. When the quaternion is expressed in a normalized form, *i.e.* with a unit quaternion, it is also referred to as Euler parameters (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008).

Unit quaternion provides a singularity-free representation of attitude kinematic, but the physical meaning of the quaternion is obscure and not as intuitive as Euler rotation angles. Moreover, quaternions provide a convenient product rule for successive rotations and a simple form of kinematic. Hence, this thesis exploits a unit quaternion notation to describe the spacecraft attitude dynamics and kinematics and to design the FDI and FDD systems. The construction of the unit quaternion arises from the observation that the rotation of coordinate systems can be uniquely described by a unit vector $\mathbf{e} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T$ giving an fixed axis of rotation as well as its sense, and an angle of rotation ϕ. The unit quaternion $\bar{\mathbf{q}}$ has four parameters defined as

$$\begin{aligned} q_1 &= e_1 \sin\frac{\phi}{2} \\ q_2 &= e_2 \sin\frac{\phi}{2} \\ q_3 &= e_3 \sin\frac{\phi}{2} \\ q_4 &= \cos\frac{\phi}{2} \end{aligned} \tag{2.9}$$

with unitary norm. Hence, the quaternion set $\bar{\mathbf{q}}$ is a vector of four parameters $\begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$ defined as

$$\bar{\mathbf{q}} = iq_1 + jq_2 + kq_3 + q_4$$

where i, j and k are hyper imaginary numbers satisfying the conditions

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ij = -ji &= k \\ jk = -kj &= i \\ ki = -ik &= j \end{aligned}$$

The overall quaternion vector $\bar{\mathbf{q}}$ can also be decomposed in a vector part, $\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T \in \mathbb{R}^3$ and a scalar part $q_4 \in \mathbb{R}$. Thus the quaternion may be written as

$$\bar{\mathbf{q}} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ q_4 \end{bmatrix} \tag{2.10}$$

The conjugate $\bar{\mathbf{q}}^*$ of the attitude unit quaternion $\bar{\mathbf{q}}$, corresponding to the inverse rotation, is defined as

$$\bar{\mathbf{q}}^* = \begin{bmatrix} -\mathbf{q} \\ q_4 \end{bmatrix}$$

The norm of quaternion $\bar{\mathbf{q}}$ is defined as

$$|\bar{\mathbf{q}}| = \sqrt{\bar{\mathbf{q}}^* \bar{\mathbf{q}}} = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} = 1$$

Now, it is convenient to introduce the matrix notation for the vector cross product for two generic vectors, which is defined by $\mathbf{u} \times \mathbf{v} = S(\mathbf{u})\mathbf{v}$, where $S(\mathbf{u})$ is a vector skew-symmetric matrix defined as

$$S(\mathbf{u}) = -S(-\mathbf{u}) = -S^T(\mathbf{u}) = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}$$

with the following properties:

$$\begin{gathered} S^T(\mathbf{x}) = -S^T(\mathbf{x}) \\ (S^2(\mathbf{x}))^T = S^2(\mathbf{x}) \\ S^T(\mathbf{x})\mathbf{y} = \mathbf{x} \times \mathbf{y} \\ S(\mathbf{x})\mathbf{y} = -S(\mathbf{y})\mathbf{x} \end{gathered}$$

Hence, defined the skew-symmetric matrix $S(\mathbf{q})$ as

$$S(\mathbf{q}) = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$$

the quaternion multiplication $\bar{\mathbf{p}} \otimes \bar{\mathbf{q}}$ of two unit quaternions $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$ is defined as

$$\bar{\mathbf{r}} = \bar{\mathbf{p}} \otimes \bar{\mathbf{q}} = \begin{bmatrix} p_4\mathbf{q} + q_4\mathbf{p} + S(\mathbf{p})\mathbf{q} \\ p_4 q_4 - \mathbf{p}^T\mathbf{q} \end{bmatrix} \tag{2.11}$$

where also the resulting $\bar{\mathbf{r}}$ is a unit quaternion and the multiplication is not a commutative operation. If $\bar{\mathbf{p}}$ describes a first rotation and $\bar{\mathbf{q}}$ a second one, then $\bar{\mathbf{r}}$ describes the overall rotation. As a consequence, the attitude difference/error of the unit quaternion $\bar{\mathbf{q}}$ with respect to $\bar{\mathbf{p}}$ can be defined as

$$\bar{\mathbf{q}}_e = \bar{\mathbf{p}}^* \otimes \bar{\mathbf{q}} = \begin{bmatrix} p_4\mathbf{q} - q_4\mathbf{p} - S(\mathbf{p})\mathbf{q} \\ p_4 q_4 + \mathbf{p}^T\mathbf{q} \end{bmatrix} \tag{2.12}$$

The coordinate transformation matrix in terms of the quaternion is defined as

$$\mathbf{R}(\bar{\mathbf{q}}) = (q_4^2 - \mathbf{q}^T\mathbf{q})\mathbf{I}_3 + 2\mathbf{q}\mathbf{q}^T - 2q_4 S(\mathbf{q}) \tag{2.13}$$

Therefore, the coordinate transformation matrix $\mathbf{R}_o^b(\bar{\mathbf{q}})$ from the orbital reference frame to the body-fixed reference frame is defined as

$$\mathbf{R}_o^b(\bar{\mathbf{q}}) = \begin{bmatrix} i_o^b & j_o^b & k_o^b \end{bmatrix} = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{2.14}$$

where i_o^b, j_o^b and k_o^b are the direction cosines (unit vectors) of the axes of the orbital frame projected on the coordinates of the body frame and parameterized by the quaternion $\bar{\mathbf{q}}$. This is the contrary rotation of the one defined by the attitude matrix $\mathbf{R}_b^o(\bar{\mathbf{q}})$, and it is used for vector transformations. Since the rotation matrices are orthogonal, it results that

$$\mathbf{R}_b^o(\bar{\mathbf{q}}) = (\mathbf{R}_o^b(\bar{\mathbf{q}}))^T = (\mathbf{R}_o^b(\bar{\mathbf{q}}))^{-1}$$
$$\mathbf{R}_b^o(\bar{\mathbf{q}})(\mathbf{R}_b^o(\bar{\mathbf{q}}))^T = (\mathbf{R}_b^o(\bar{\mathbf{q}}))^T\mathbf{R}_b^o(\bar{\mathbf{q}}) = \mathbf{I}_3$$

Therefore, this gives that the coordinate transformation matrix $\mathbf{R}_b^o(\bar{\mathbf{q}})$ from the body-fixed reference frame to the orbital reference frame is defined as

$$\mathbf{R}_b^o(\bar{\mathbf{q}}) = \begin{bmatrix} i_b^o & j_b^o & k_b^o \end{bmatrix}^T = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 - q_3q_4) & 2(q_1q_3 + q_2q_4) \\ 2(q_1q_2 + q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 - q_1q_4) \\ 2(q_1q_3 - q_2q_4) & 2(q_2q_3 + q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{2.15}$$

where i_b^o, j_b^o and k_b^o are the direction cosines (unit vectors) of the axes of the body frame projected on the coordinates of the orbital frame and parameterized by the attitude quaternion $\bar{\mathbf{q}}$. This matrix describes the attitude of the satellite with respect to the orbital reference frame. Moreover, considering the definition of $\mathbf{R}(\bar{\mathbf{q}})$ in Eq. (2.13), it results that

$$\mathbf{R}(\bar{\mathbf{q}}) = \mathbf{R}(-\bar{\mathbf{q}})$$

Hence, the same attitude can be described by two quaternions $\bar{\mathbf{q}}$ and $-\bar{\mathbf{q}}$, the first one given when the angle of rotation is ϕ, the latter for the angle $2\pi+\phi$. This implies that the unit quaternion inherently possesses a sign ambiguity. Considering the definition of Eq. (2.12) of the error unit quaternion, this ambiguity leads to having two choices, the attitude error of $\bar{\mathbf{q}}$ with respect to $\bar{\mathbf{p}}$ and the error of $\bar{\mathbf{q}}$ with respect to $-\bar{\mathbf{p}}$, where one exhibits a geodesic rotation error ($\bar{\mathbf{p}}^* \otimes \bar{\mathbf{q}}$) while the other a non-geodesic rotation error ($-\bar{\mathbf{p}}^* \otimes \bar{\mathbf{q}}$).

Finally, quaternions provide simple methods for calculation of successive rotations. Let the quaternion $\bar{\mathbf{q}}_a^b$ describing the rotation of the reference frame F_b in the reference frame F_a and the quaternion $\bar{\mathbf{q}}_b^c$ describing the rotation of the reference frame F_c in the reference frame F_b be given. Then, the product quaternion in Eq. (2.11) provides an elegant method for calculation of the total transformation from the frame F_a to the frame F_c:

$$\bar{\mathbf{q}}_a^c = \mathbf{Q}(\bar{\mathbf{q}}_a^b)\bar{\mathbf{q}}_b^c$$

where

$$\mathbf{Q}(\bar{\mathbf{q}}) = \begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix}$$

In addition, $\bar{\mathbf{q}}$ can also be expressed in terms of $\mathbf{R}(\bar{\mathbf{q}})$ as

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \frac{1}{4q_4}\begin{bmatrix} R_{23}-R_{32} \\ R_{31}-R_{13} \\ R_{12}-R_{21} \end{bmatrix}$$
$$q_4 = \pm\frac{1}{2}\sqrt{1+tr(\mathbf{R})}$$

The algorithm to transform a set of attitude parameters from Euler angles to unit quaternion, or from unit quaternion to Euler angles can be found in (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997), exploiting the fact that $\mathbf{R}(\Theta)=\mathbf{R}(\bar{\mathbf{q}})$ to derive the conversion relations, where $\mathbf{R}$ are the rotation matrices previously defined.

Again, to avoid misunderstandings, the subscripted quaternion vector $\bar{\mathbf{q}}_o = \begin{bmatrix} q_{1,o} & q_{2,o} & q_{3,o} & q_{4,o} \end{bmatrix}^T$ will represent the spacecraft attitude with respect to the orbital reference frame, whereas the subscripted quaternion vector $\bar{\mathbf{q}} = \begin{bmatrix} q_{1,i} & q_{2,i} & q_{3,i} & q_{4,i} \end{bmatrix}^T$ will represent the spacecraft attitude with respect to the inertial reference frame.

2.3 Angular Velocity

The angular velocity $\boldsymbol{\omega}$ is used to investigate the angular displacements that occur over time. Angular velocity is dependent on the frame of reference, and is expressed in this thesis with the notation $\boldsymbol{\omega}_{ab}^c$, which defines the rotation rate of the frame F_a with respect to F_b as seen by F_c. Angular velocities add, but only when they are in the same reference frame. For example, the following relation between the relative angular velocities of the body-fixed, orbital and inertial reference frames is valid:

$$\boldsymbol{\omega}_{ib}^{b} = \boldsymbol{\omega}_{ob}^{b} + \boldsymbol{\omega}_{io}^{b} \tag{2.16}$$

where the resulting angular velocity is seen by the body frame, and $\boldsymbol{\omega}_{ib}^{b}$, $\boldsymbol{\omega}_{ob}^{b}$ and $\boldsymbol{\omega}_{io}^{b}$ are the angular velocity vectors of the satellite from body to inertial frame, from body to orbital frame and from orbit to inertial frame as seen in the body-fixed reference frame, respectively.

When the angular velocities are expressed in different reference frames, however, it is necessary to perform transformations. For example, this is evident in the relation

$$\boldsymbol{\omega}_{ib}^{b} = \mathbf{R}_{o}^{b}\boldsymbol{\omega}_{ob}^{o} + \mathbf{R}_{i}^{b}\boldsymbol{\omega}_{io}^{i} = \boldsymbol{\omega}_{ob}^{b} + \mathbf{R}_{o}^{b}\boldsymbol{\omega}_{io}^{o} \tag{2.17}$$

where the resulting angular velocity is seen by the body frame, and $\boldsymbol{\omega}_{ob}^{o}$, $\boldsymbol{\omega}_{io}^{i}$ and $\boldsymbol{\omega}_{io}^{o}$ are the angular velocity vectors of the satellite from body to orbital frame as seen in the orbital frame, from orbit to inertial frame as seen in the inertial frame and from orbit to inertial frame as seen in the orbital frame. Hence, considering the second part of Eq. (2.17), the following angular velocity vectors can be defined:

$$\boldsymbol{\omega}_{ib}^{b} = \begin{bmatrix} \omega_{x,i} \\ \omega_{y,i} \\ \omega_{z,i} \end{bmatrix} \qquad \boldsymbol{\omega}_{ob}^{b} = \begin{bmatrix} \omega_{x,o} \\ \omega_{y,o} \\ \omega_{z,o} \end{bmatrix} \qquad \boldsymbol{\omega}_{io}^{o} = \begin{bmatrix} 0 \\ -\omega_{o} \\ 0 \end{bmatrix} \tag{2.18}$$

where ω_o is the angular velocity of the spacecraft orbit, and for a circular equatorial orbit is assumed constant and calculated from relation

$$\omega_o^2 = \frac{GM}{R^3} = \frac{\mu}{R^3} \tag{2.19}$$

where R is the orbit radius (*i.e.* the distance from the centre of the Earth to the satellite's centre of gravity), which is constant for a circular orbit, G is Newton's specific gravity constant, M is the Earth mass and $\mu = 3.986 \cdot 10^{14}\ \mathrm{m}^3/\mathrm{s}^2$ is the Earth's gravitational coefficient.

2.4 Equations of Motion

The mathematical model of a satellite is described by nonlinear dynamic and kinematic differential equations of motion (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008). The dynamics relates torques acting on the satellite to the satellite angular velocity in the inertial reference frame. The kinematics provides integration of the angular velocity $\boldsymbol{\omega}_{ib}^{b}$ between the inertial and body-fixed reference frames, as seen in the body reference frame. The attitude is parameterized by four components of the quaternion $\bar{\mathbf{q}}_i = \begin{bmatrix} q_{1,i} & q_{2,i} & q_{3,i} & q_{4,i} \end{bmatrix}^T$ describing the rotation of the body-fixed frame in the inertial reference frame.

2.4.1 Spacecraft Inertia Matrix

The inertia matrix of the satellite is a constant real symmetric matrix given by

$$\mathbf{I}_{sat} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \tag{2.20}$$

If the physical body has a mass symmetrically distributed about its axes, or assumed to be symmetrically distributed, the inertia matrix becomes diagonal:

$$\mathbf{I}_{sat} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{2.21}$$

where I_{xx}, I_{yy} and I_{zz} in Eqs. (2.20) and (2.21) are the moment of inertia about the body axes x_b, y_b and z_b, and they are calculated by

$$I_{xx} = \int_m (y^2 + z^2) dm$$
$$I_{yy} = \int_m (x^2 + z^2) dm$$
$$I_{zz} = \int_m (x^2 + y^2) dm$$

whereas the inertia products in Eq. (2.20) are defined as

$$I_{xy} = I_{yx} = \int_m xydm$$
$$I_{xz} = I_{zx} = \int_m xzdm$$
$$I_{yz} = I_{zy} = \int_m yzdm$$

The integrations are performed over the total mass of the body and x, y and z represent the components of the position vector of each particle of mass dm with respect to the centre of mass of the body. The terms I_{xx}, I_{yy} and I_{zz} are called principal moments of inertia when the matrix is diagonal, *i.e.* if the mass distribution is symmetric with respect to the (principal) axes of the body. The moments of inertia of a body about particular axes are measures of the distribution of the mass about those axes, and thus more they are small more the mass is concentrated.

2.4.2 Dynamic Equations of Motion

Some assumptions are made for the dynamic modelling of the satellite. For example, the satellite is assumed to act as a rigid body and behave as a point mass model for orbital dynamics. The dynamics of the satellite can be derived by using a Newton-Euler formulation for the rigid body (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008), where the equations of motion are derived from the definition of angular momentum $h = I\omega$. The overall angular momentum of a spacecraft changes according to the applied torques. This leads to the following model expressed in the body frame:

$$\mathbf{I}_{sat}\dot{\boldsymbol{\omega}}_{ib}^{b} + \boldsymbol{\omega}_{ib}^{b} \times (\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^{b}) = \mathbf{T}_{tot}^{b} \tag{2.22}$$

With a simpler notation, equation (2.22) becomes

$$\mathbf{I}_{sat}\dot{\boldsymbol{\omega}}_{ib}^{b} + S(\boldsymbol{\omega}_{ib}^{b})\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^{b} = \mathbf{T}_{tot}^{b} \tag{2.23}$$

where the cross product is written as a matrix operation using the skew-symmetric matrix

$$S(\boldsymbol{\omega}_{ib}^{b}) = \begin{bmatrix} 0 & -\omega_{z,i} & \omega_{y,i} \\ \omega_{z,i} & 0 & -\omega_{x,i} \\ -\omega_{y,i} & \omega_{x,i} & 0 \end{bmatrix}$$

The term $\mathbf{T}_{tot}^{b}$ represents the total torques about the mass centre of the spacecraft. Eq. (2.22) represents the Euler equation of moment. Differentiating the Eq. (2.17) gives

$$\dot{\boldsymbol{\omega}}_{ib}^{b} = \dot{\boldsymbol{\omega}}_{ob}^{b} - S(\boldsymbol{\omega}_{ob}^{b})\mathbf{R}_{o}^{b}\boldsymbol{\omega}_{io}^{o} \tag{2.24}$$

and

$$\dot{\boldsymbol{\omega}}_{ob}^{b} = \dot{\boldsymbol{\omega}}_{ib}^{b} + S(\boldsymbol{\omega}_{ob}^{b})\mathbf{R}_{o}^{b}\boldsymbol{\omega}_{io}^{o} \tag{2.25}$$

using the skew-symmetric matrix

$$S(\boldsymbol{\omega}_{ob}^{b}) = \begin{bmatrix} 0 & -\omega_{z,o} & \omega_{y,o} \\ \omega_{z,o} & 0 & -\omega_{x,o} \\ -\omega_{y,o} & \omega_{x,o} & 0 \end{bmatrix}$$

Considering the variation of total angular moment of the body as the sum of control torques $\mathbf{T}_{ctrl}^{b}$ allocated to control the satellite and external disturbance torques $\mathbf{T}_{ext}^{b}$ acting on the satellite, Eq. (2.23) can be written as

$$\mathbf{I}_{sat}\dot{\boldsymbol{\omega}}_{ib}^{b} = -S(\boldsymbol{\omega}_{ib}^{b})\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^{b} + \mathbf{T}_{ctrl}^{b} + \mathbf{T}_{ext}^{b} \tag{2.26}$$

where the total torque about the mass centre is broken up into two distinct terms:

$$\mathbf{T}_{tot}^{b} = \mathbf{T}_{ctrl}^{b} + \mathbf{T}_{ext}^{b} \tag{2.27}$$

Solving the Eq. (2.26) for $\dot{\boldsymbol{\omega}}_{ib}^{b}$ leads to

$$\dot{\boldsymbol{\omega}}_{ib}^{b} = -\mathbf{I}_{sat}^{-1}S(\boldsymbol{\omega}_{ib}^{b})\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^{b} + \mathbf{I}_{sat}^{-1}(\mathbf{T}_{ctrl}^{b} + \mathbf{T}_{ext}^{b}) = \begin{bmatrix} -\dfrac{(I_{zz}-I_{yy})}{I_{xx}}\omega_{y}\omega_{z} + \dfrac{(T_{ctrl,1}^{b}+T_{ext,1}^{b})}{I_{xx}} \\ -\dfrac{(I_{xx}-I_{zz})}{I_{yy}}\omega_{x}\omega_{z} + \dfrac{(T_{ctrl,2}^{b}+T_{ext,2}^{b})}{I_{yy}} \\ -\dfrac{(I_{yy}-I_{xx})}{I_{zz}}\omega_{x}\omega_{y} + \dfrac{(T_{ctrl,3}^{b}+T_{ext,3}^{b})}{I_{zz}} \end{bmatrix} \tag{2.28}$$

when principal body axes are considered. These equations are known as Euler's equations of motion for the rigid body.

2.4.3 Kinematic Equations of Motion

The kinematics of the spacecraft is derived by integrating the angular velocity of the satellite. It is used to describe the orientation of the satellite body frame with respect to another reference frame (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997; Egeland and Gravdahl 2002; Tewari 2007; Wie 2008). The kinematic differential equation between the spacecraft angular velocity and the unit quaternion exploited in this dissertation are given by

$$\dot{\bar{\mathbf{q}}}_i = \begin{bmatrix} \dot{\mathbf{q}}_i \\ \dot{q}_{4,i} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_{4,i} & -q_{3,i} & q_{2,i} \\ q_{3,i} & q_{4,i} & -q_{1,i} \\ -q_{2,i} & q_{1,i} & q_{4,i} \\ -q_{1,i} & -q_{2,i} & -q_{3,i} \end{bmatrix} \begin{bmatrix} \omega_{x,i} \\ \omega_{y,i} \\ \omega_{z,i} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_{4,i}\mathbf{I}_3 + S(\mathbf{q}_i) \\ -\mathbf{q}_i^T \end{bmatrix} \boldsymbol{\omega}_{ib}^b$$

or, with an alternative and more compact notation by defining the skew symmetric matrix $\Omega(\boldsymbol{\omega}_{ib}^b)$, as

$$\dot{\bar{\mathbf{q}}}_i = \begin{bmatrix} \dot{\mathbf{q}}_i \\ \dot{q}_{4,i} \end{bmatrix} = \frac{1}{2} \Omega(\boldsymbol{\omega}_{ib}^b) \bar{\mathbf{q}}_i \tag{2.29}$$

with

$$\Omega(\boldsymbol{\omega}_{ib}^b) = \begin{bmatrix} 0 & \omega_{z,i} & -\omega_{y,i} & \omega_{x,i} \\ -\omega_{z,i} & 0 & \omega_{x,i} & \omega_{y,i} \\ \omega_{y,i} & -\omega_{x,i} & 0 & \omega_{z,i} \\ -\omega_{x,i} & -\omega_{y,i} & -\omega_{z,i} & 0 \end{bmatrix}$$

or, in an explicit way, as

$$\begin{aligned} \dot{q}_{1,i} &= \frac{1}{2}(\omega_{x,i} q_{4,i} - \omega_{y,i} q_{3,i} + \omega_{z,i} q_{2,i}) \\ \dot{q}_{2,i} &= \frac{1}{2}(\omega_{x,i} q_{3,i} + \omega_{y,i} q_{4,i} - \omega_{z,i} q_{1,i}) \\ \dot{q}_{3,i} &= \frac{1}{2}(-\omega_{x,i} q_{2,i} + \omega_{y,i} q_{1,i} + \omega_{z,i} q_{4,i}) \\ \dot{q}_{4,i} &= \frac{1}{2}(-\omega_{x,i} q_{1,i} - \omega_{y,i} q_{2,i} - \omega_{z,i} q_{3,i}) \end{aligned} \tag{2.30}$$

In particular, in Eq. (2.29) there are described the kinematic relations between the spacecraft angular velocity of the body-fixed frame with respect the inertial reference frame, expressed in the body frame, and the unit quaternion representing the body attitude with respect to the inertial reference frame, expressed in the body frame.

In a similar way, the kinematic equations can be also written using the attitude and angular velocity of the body expressed with respect to the orbital reference frame. The conversion relations between the angular velocity of the body frame with respect to the orbital frame or inertial frame respectively, and expressed in the body frame, are defined as

$$\boldsymbol{\omega}_{ob}^{b} = \boldsymbol{\omega}_{ib}^{b} - \boldsymbol{\omega}_{io}^{b} = \boldsymbol{\omega}_{ib}^{b} - \mathbf{R}_{o}^{b}\boldsymbol{\omega}_{io}^{o} = \boldsymbol{\omega}_{ib}^{b} + \omega_o j_o^b = \boldsymbol{\omega}_{ib}^{b} + \omega_o \begin{bmatrix} 2(q_{1,o}q_{2,o} + q_{3,o}q_{4,o}) \\ (1 - 2q_{1,o}^2 - 2q_{3,o}^2) \\ 2(q_{2,o}q_{3,o} - q_{1,o}q_{4,o}) \end{bmatrix} \tag{2.31}$$

$$\boldsymbol{\omega}_{ib}^{b} = \boldsymbol{\omega}_{ob}^{b} + \boldsymbol{\omega}_{io}^{b} = \boldsymbol{\omega}_{ob}^{b} + \mathbf{R}_{o}^{b}\boldsymbol{\omega}_{io}^{o} = \boldsymbol{\omega}_{ob}^{b} - \omega_o j_o^b = \boldsymbol{\omega}_{ob}^{b} - \omega_o \begin{bmatrix} 2(q_{1,o}q_{2,o} + q_{3,o}q_{4,o}) \\ (1 - 2q_{1,o}^2 - 2q_{3,o}^2) \\ 2(q_{2,o}q_{3,o} - q_{1,o}q_{4,o}) \end{bmatrix} \tag{2.32}$$

where

$$\boldsymbol{\omega}_{io}^{o} = \begin{bmatrix} 0 \\ -\omega_o \\ 0 \end{bmatrix} \qquad j_o^b = \begin{bmatrix} 2(q_{1,o}q_{2,o} + q_{3,o}q_{4,o}) \\ (1 - 2q_{1,o}^2 - 2q_{3,o}^2) \\ 2(q_{2,o}q_{3,o} - q_{1,o}q_{4,o}) \end{bmatrix}$$

and ω_o is the angular velocity of the spacecraft orbit. Therefore, by considering the satellite angular velocity and the attitude with respect to the orbital frame, Eq. (2.29) can be rewritten as

$$\dot{\bar{\mathbf{q}}}_{orb} = \begin{bmatrix} \dot{\mathbf{q}}_{orb} \\ \dot{q}_{4,orb} \end{bmatrix} = \frac{1}{2}\Omega(\boldsymbol{\omega}_{ob}^{b})\bar{\mathbf{q}}_{orb} = \frac{1}{2}\Omega(\boldsymbol{\omega}_{ib}^{b}, \omega_o)\bar{\mathbf{q}}_{orb} \tag{2.33}$$

with, considering a circular equatorial orbit,

$$\Omega(\boldsymbol{\omega}_{ib}^{b}, \omega_o) = \begin{bmatrix} 0 & \omega_z & -\omega_y + \omega_o & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y + \omega_o \\ \omega_y - \omega_o & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y - \omega_o & -\omega_z & 0 \end{bmatrix}$$

or, in an explicit way exploiting the relations of Eq. (2.31), written as

$$\begin{aligned} \dot{q}_{1,o} &= \frac{1}{2}(\omega_{x,o}q_{4,o} - \omega_{y,o}q_{3,o} + \omega_{z,o}q_{2,o}) = \frac{1}{2}(\omega_{x,i}q_{4,o} - \omega_{y,i}q_{3,o} + \omega_{z,i}q_{2,o} + \omega_o q_{3,o}) \\ \dot{q}_{2,o} &= \frac{1}{2}(\omega_{x,o}q_{3,o} + \omega_{y,o}q_{4,o} - \omega_{z,o}q_{1,o}) = \frac{1}{2}(\omega_{x,i}q_{3,o} + \omega_{y,i}q_{4,o} - \omega_{z,i}q_{1,o} + \omega_o q_{4,o}) \\ \dot{q}_{3,o} &= \frac{1}{2}(-\omega_{x,o}q_{2,o} + \omega_{y,o}q_{1,orb} + \omega_{z,o}q_{4,o}) = \frac{1}{2}(-\omega_{x,i}q_{2,o} + \omega_{y,i}q_{1,o} + \omega_{z,i}q_{4,o} - \omega_o q_{1,o}) \\ \dot{q}_{4,o} &= \frac{1}{2}(-\omega_{x,o}q_{1,o} - \omega_{y,o}q_{2,o} - \omega_{z,o}q_{3,o}) = \frac{1}{2}(-\omega_{x,i}q_{1,o} - \omega_{y,i}q_{2,o} - \omega_{z,i}q_{3,o} - \omega_o q_{2,o}) \end{aligned} \tag{2.34}$$

2.5 Environmental Disturbance Torques

In order to design the attitude control system and fault diagnosis system, environmental disturbance torques usually acting on the spacecraft must be modelled sufficiently. The torques must be modelled as a function of time, spacecraft position and attitude so that they can be integrated to Euler's equations and any other mathematical models.

Spacecraft in orbit encounter small disturbance torques from various environmental sources. These torques are either secular, which accumulate over time, or cyclic, which vary sinusoidal over an orbit. Both types are discussed in Kaplan (1976), Wertz (1978), Hughes (1986) and Sidi (1997). Different environmental torques are more prevalent at different altitudes. The relationship between altitude and disturbance torque strength is shown in Fig. 2.4.

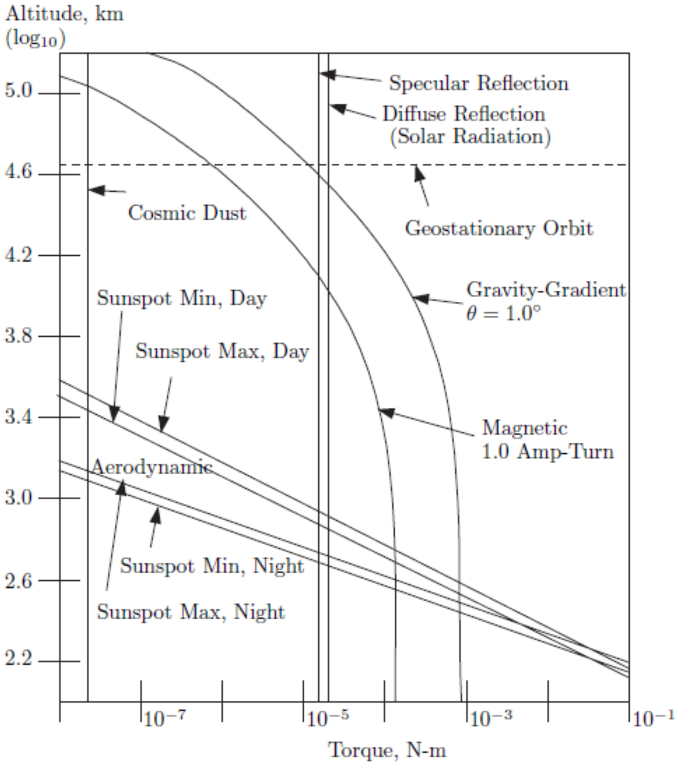


Figure 2.4 - Comparison of Environmental Torques (adapted from Hughes 1986).

In Low Earth Orbits (LEO) the dominant sources of environmental disturbance torques are the solar radiation pressure, aerodynamic drag and Earth's gravitational and magnetic fields. There are also internal torques primarily resulted from internal moving hardware, propellant leakage, thrust misalignment and so on.

The solar radiation pressure is effective on attitude of the satellite for altitudes higher than 1000 km. The gravity gradient disturbance are most significant below 1000 km. Aerodynamic perturbations are most effective below 500 km and negligible over 1000 km altitudes.

In this thesis, due to the altitude of the LEO satellite considered in the simulations (350 km), only gravity-gradient and aerodynamic torques are considered. Hence, the total torques, control torques and external disturbance torques are represented by

$$\mathbf{T}_{tot}^{b} = \mathbf{T}_{ctrl}^{b} + \mathbf{T}_{ext}^{b} = \mathbf{T}_{ctrl}^{b} + \mathbf{T}_{gg}^{b} + \mathbf{T}_{aero}^{b} \tag{2.35}$$

which are all expressed in the satellite body frame.

2.5.1 Gravity-Gradient Torque

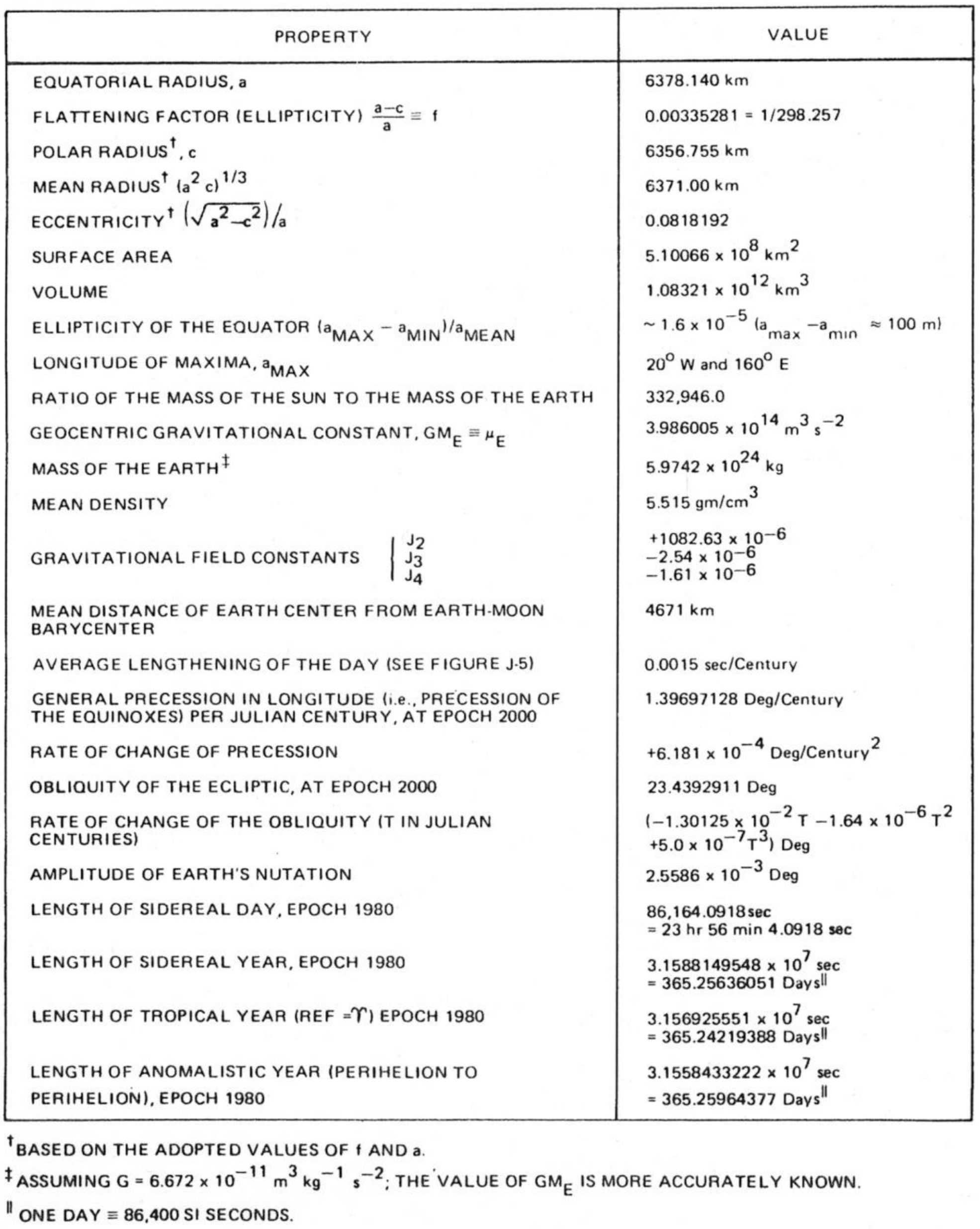
Any non-symmetrical object in the orbit is affected by a gravitational torque because of the variation in the Earth's gravitational force over the object. There are many mathematical models for gravity gradient torque (Kaplan 1976; Wertz 1978; Hughes 1986; Sidi 1997). The most common one is derived by assuming homogeneous mass distribution of the Earth. Since the gravitational force field varies with the inverse square of the distance from the Earth, there is a greater force on the portion of the spacecraft closer to the Earth. The gravity-gradient torque is constant for Earth-oriented spacecraft and cyclic for inertial oriented vehicles. It is mainly influenced by the moment of inertia of the spacecraft and the altitude of the orbit. The gravity-gradient torque $\mathbf{T}_{gg}^{b}$ can be modelled, according to Wertz (1978) and Sidi (1997), as

$$\mathbf{T}_{gg}^{b} = 3\omega_o^2 (k_o^b \times \mathbf{I}_{sat} k_o^b) = \frac{3\mu}{R^3}(k_o^b \times \mathbf{I}_{sat} k_o^b) \tag{2.36}$$

where $\mathbf{I}_{sat}$ is the inertia matrix of the spacecraft, k_o^b is the directional cosine from the matrix $\mathbf{R}_o^b$ describing the attitude of the spacecraft with respect to the orbital reference frame and ω_o is the angular velocity of the satellite orbit, as already stated, and for a circular orbit it is assumed constant and calculated from relation

$$\omega_o^2 = \frac{GM}{R^3} = \frac{\mu}{R^3} \tag{2.37}$$

where R is the orbit radius (*i.e.* the distance from the centre of the Earth to the satellite's centre of gravity), which is constant for a circular orbit, G is Newton's specific gravity constant, M is the Earth mass and $\mu = 3.986 \cdot 10^{14}\ \mathrm{m^3/s^2}$ is the Earth's gravitational coefficient. Tab. 2.2 summarizes the physical properties of the Earth (Wertz 1978).



PROPERTY	VALUE
EQUATORIAL RADIUS, a	6378.140 km
FLATTENING FACTOR (ELLIPTICITY) $\frac{a-c}{a} \equiv f$	0.00335281 = 1/298.257
POLAR RADIUS†, c	6356.755 km
MEAN RADIUS† $(a^2 c)^{1/3}$	6371.00 km
ECCENTRICITY† $\left(\sqrt{a^2-c^2}\right)/a$	0.0818192
SURFACE AREA	5.10066×10^8 km^2
VOLUME	1.08321×10^{12} km^3
ELLIPTICITY OF THE EQUATOR $(a_{MAX} - a_{MIN})/a_{MEAN}$	$\sim 1.6 \times 10^{-5}$ $(a_{max} - a_{min} \approx 100$ m)
LONGITUDE OF MAXIMA, a_{MAX}	20° W and 160° E
RATIO OF THE MASS OF THE SUN TO THE MASS OF THE EARTH	332,946.0
GEOCENTRIC GRAVITATIONAL CONSTANT, $GM_E \equiv \mu_E$	3.986005×10^{14} m^3 s^{-2}
MASS OF THE EARTH‡	5.9742×10^{24} kg
MEAN DENSITY	5.515 gm/cm^3
GRAVITATIONAL FIELD CONSTANTS J_2 J_3 J_4	$+1082.63 \times 10^{-6}$ -2.54×10^{-6} -1.61×10^{-6}
MEAN DISTANCE OF EARTH CENTER FROM EARTH-MOON BARYCENTER	4671 km
AVERAGE LENGTHENING OF THE DAY (SEE FIGURE J-5)	0.0015 sec/Century
GENERAL PRECESSION IN LONGITUDE (i.e., PRECESSION OF THE EQUINOXES) PER JULIAN CENTURY, AT EPOCH 2000	1.39697128 Deg/Century
RATE OF CHANGE OF PRECESSION	$+6.181 \times 10^{-4}$ Deg/Century^2
OBLIQUITY OF THE ECLIPTIC, AT EPOCH 2000	23.4392911 Deg
RATE OF CHANGE OF THE OBLIQUITY (T IN JULIAN CENTURIES)	$(-1.30125 \times 10^{-2}\,T - 1.64 \times 10^{-6}\,T^2$ $+5.0 \times 10^{-7}\,T^3)$ Deg
AMPLITUDE OF EARTH'S NUTATION	2.5586×10^{-3} Deg
LENGTH OF SIDEREAL DAY, EPOCH 1980	86,164.0918 sec = 23 hr 56 min 4.0918 sec
LENGTH OF SIDEREAL YEAR, EPOCH 1980	3.1588149548×10^7 sec = 365.25636051 Days‖
LENGTH OF TROPICAL YEAR (REF = ♈) EPOCH 1980	3.156925551×10^7 sec = 365.24219388 Days‖
LENGTH OF ANOMALISTIC YEAR (PERIHELION TO PERIHELION), EPOCH 1980	3.1558433222×10^7 sec = 365.25964377 Days‖

†BASED ON THE ADOPTED VALUES OF f AND a.

‡ASSUMING G = 6.672×10^{-11} m^3 kg^{-1} s^{-2}; THE VALUE OF GM_E IS MORE ACCURATELY KNOWN.

‖ ONE DAY ≡ 86,400 SI SECONDS.

Table 2.2 - Physical properties of Earth (Wertz 1978).

The unit vector k_o^b corresponds to the zenith, *i.e.* a unit vector in the body coordinate system along the line connecting the satellite centre of gravity and the Earth centre pointing away from the Earth:

$$k_o^b = \begin{bmatrix} 2(q_{1,o}q_{3,o} - q_{2,o}q_{4,o}) \\ 2(q_{2,o}q_{3,o} + q_{1,o}q_{4,o}) \\ 1 - 2q_{1,o}^2 - 2q_{2,o}^2 \end{bmatrix}$$

Then, Eq. (2.36) becomes

$$\mathbf{T}_{gg}^b = \frac{3\mu}{R^3}\begin{bmatrix} 2(I_{zz} - I_{yy})(q_{2,o}q_{3,o} + q_{1,o}q_{4,o})(1 - 2q_{1,o}^2 - 2q_{2,o}^2) \\ 2(I_{xx} - I_{zz})(q_{1,o}q_{3,o} - q_{2,o}q_{4,o})(1 - 2q_{1,o}^2 - 2q_{2,o}^2) \\ 4(I_{yy} - I_{xx})(q_{2,o}q_{3,o} + q_{1,o}q_{4,o})(q_{1,o}q_{3,o} - q_{2,o}q_{4,o}) \end{bmatrix} \tag{2.38}$$

2.5.2 Aerodynamic Torque

This disturbance is most effective on satellites orbiting below 400-500 km. The drag force created by the air molecules interacting with satellite body produce a torque on the satellite, moreover reducing its velocity and resulting in a lower orbit for the satellite.

An approximation of the satellite structure by a collection of simple geometrical figures has been assumed. In particular, in this thesis the considered satellite is assumed to be a simple rectangular cuboid, *i.e.* a regular parallelepiped with six rectangular faces. The aerodynamic force on each element, according to Wertz (1978) and Sidi (1997), is determined by integrating the following equation

$$dF_{aero} = -\frac{1}{2} C_D \rho v^2 (\hat{n} \cdot \hat{v}) dS$$

where dS is the surface element, $\hat{n}$ is an outward normal to the surface, $\hat{v}$ is the unit vector in the direction of translational velocity V, C_D is the drag coefficient, and ρ is the atmospheric density. The total aerodynamic torque is the vector sum of the torques acting on individual parts of the satellite

$$\begin{aligned} \mathbf{T}^b_{aero} &= \sum_{i=1}^{6} (c_p - c_g) \times F_{aero,i} \\ F_{aero,i} &= -\frac{1}{2} \rho V^2 C_D S_{\text{exposed}} (\hat{n}_i \cdot \hat{v}) \end{aligned}$$

where $(c_p - c_g)$ is the vector from the spacecraft centre of mass c_g to the centre of pressure c_p of the *i-th* surface element. Hence, the aerodynamic disturbance torque can be modelled as

$$\begin{aligned} \mathbf{T}^b_{aero} &= F_{aero}((c_p - c_g) \times \hat{v}) = -\frac{1}{2} \rho V^2 C_D S_{\text{exposed}} ((c_p - c_g) \times \hat{v}) \\ F_{aero} &= -\frac{1}{2} \rho V^2 C_D S_{\text{exposed}} \end{aligned} \tag{2.39}$$

or explicitly as

$$\mathbf{T}^b_{aero} = -\frac{1}{2} \rho V^2 C_D S_{\text{exposed}} \begin{bmatrix} 2(q_{1,o}q_{3,o} + q_{2,o}q_{4,o})c_{pa,y} - 2(q_{1,o}q_{2,o} - q_{3,o}q_{4,o})c_{pa,z} \\ (1 - 2q_{2,o}^2 - 2q_{3,o}^2)c_{pa,z} - 2(q_{1,o}q_{3,o} + q_{2,o}q_{4,o})c_{pa,x} \\ 2(q_{1,o}q_{2,o} - q_{3,o}q_{4,o})c_{pa,x} - (1 - 2q_{2,o}^2 - 2q_{3,o}^2)c_{pa,y} \end{bmatrix} \tag{2.40}$$

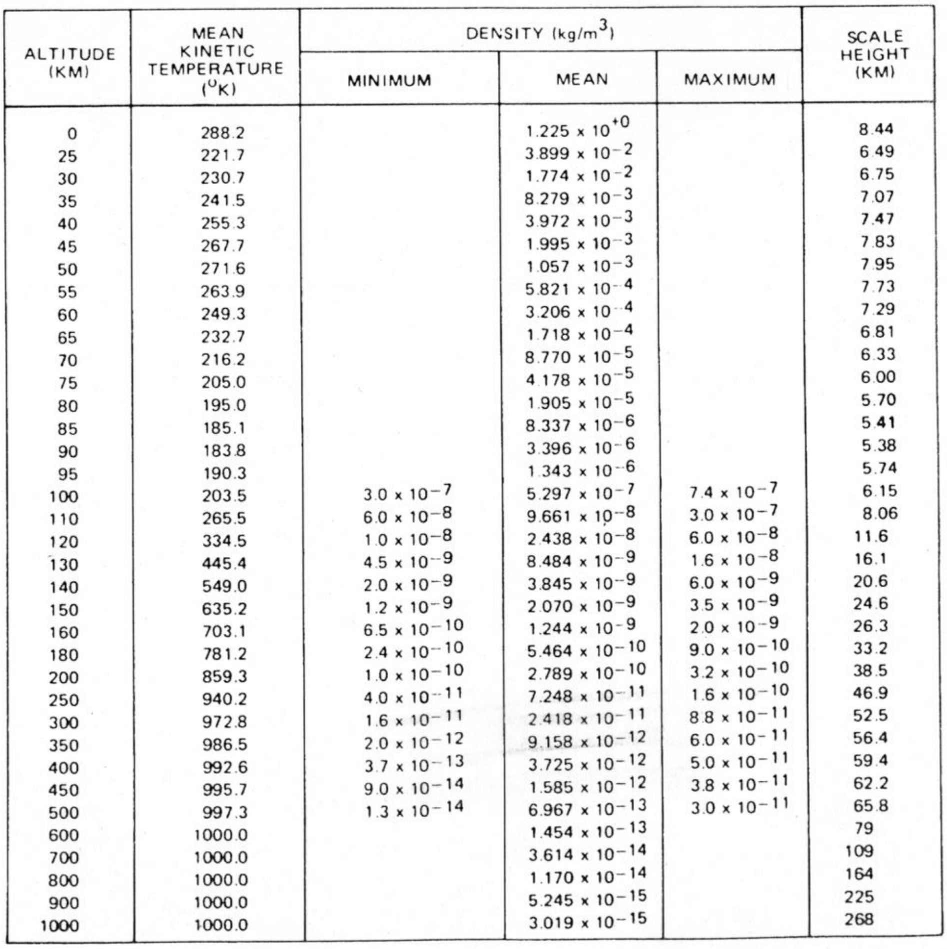
where S_{exposed} is the area exposed to the aerodynamic flux and perpendicular to the unit vector $\hat{v}$, $c_{pa} = \begin{bmatrix} c_{pa,x} & c_{pa,y} & c_{pa,z} \end{bmatrix}^T$ are the components of the displacement vector, *i.e.* the vector joining the centre of mass and the aerodynamic centre of pressure of the satellite, expressed in the body frame with origin in the centre of mass. The exposed area can be calculated by

$$\begin{aligned} S_{\text{exposed}} = &\left| dw\left(4(q_{1,o}q_{2,o} + q_{3,o}q_{4,o}) \cdot (q_{2,o}q_{3,o} + q_{1,o}q_{4,o}) - 2(q_{1,o}q_{3,o} - q_{2,o}q_{4,o}) \cdot (1 - 2q_{1,o}^2 - 2q_{3,o}^2)\right)\right| + \\ &+ \left| dh\left(2(q_{1,o}q_{2,o} + q_{3,o}q_{4,o}) \cdot (1 - 2q_{1,o}^2 - 2q_{2,o}^2) - 4(q_{1,o}q_{3,o} - q_{2,o}q_{4,o}) \cdot (q_{2,o}q_{3,o} - q_{1,o}q_{4,o})\right)\right| + \\ &+ \left| wh\left((1 - 2q_{1,o}^2 - 2q_{3,o}^2) \cdot (1 - 2q_{1,o}^2 - 2q_{2,o}^2) - 4(q_{2,o}q_{3,o} + q_{1,o}q_{4,o}) \cdot (q_{2,o}q_{3,o} - q_{1,o}q_{4,o})\right)\right| \end{aligned}$$

where *d*, *w* and *h* are the depth (x_b -direction), width (y_b -direction) and height (z_b -direction) of the satellite body respectively, expressed in the body frame. Since the satellite is assumed to be a rectangular cuboid with edges aligned with the principal axes of inertia, these coincide with the cuboid dimensions.

Now, it is worth observing that, the design of the FDI/FDD systems will exploit an explicit decoupling only of the aerodynamic disturbance torque since the gravitational disturbance, even if it can produce torques at least one order of magnitude larger than other external torques, has a model which is almost perfectly known. Thus, the effect of the gravitational disturbance does not need to be decoupled. It is worth noting that, mainly due to the presence of the unknown terms ρ and C_D, the term $F_a = -\frac{1}{2}\rho V^2 C_D S_{\text{exposed}}$ in Eqs. (2.39) and (2.40) represents the main source of uncertainty from which the diagnostic signals provided by the designed residual generators will have to be decoupled.

Tab. 2.3 summarizes the properties of the upper atmosphere of the Earth (Wertz 1978). The mean profiles between 25 and 500 km are from the COSPAR International Reference Atmosphere, CIRA 72 (1972). Between 500 and 1000 km, the CIRA 72 profile for $T_\infty = 1000\ \mathrm{K}$ was used to indicate the densities to be expected. The maximum and minimum values of air density between 100 and 500 km were extracted from the explanatory material in CIRA 72 and indicate the variation in densities which can be obtained with the models. Sea level temperature and density are from U.S. Standard Atmosphere (1976).



ALTITUDE (KM)	MEAN KINETIC TEMPERATURE (°K)	DENSITY (kg/m^3)			SCALE HEIGHT (KM)
		MINIMUM	MEAN	MAXIMUM	
0	288.2		$1.225 \times 10^{+0}$		8.44
25	221.7		3.899×10^{-2}		6.49
30	230.7		1.774×10^{-2}		6.75
35	241.5		8.279×10^{-3}		7.07
40	255.3		3.972×10^{-3}		7.47
45	267.7		1.995×10^{-3}		7.83
50	271.6		1.057×10^{-3}		7.95
55	263.9		5.821×10^{-4}		7.73
60	249.3		3.206×10^{-4}		7.29
65	232.7		1.718×10^{-4}		6.81
70	216.2		8.770×10^{-5}		6.33
75	205.0		4.178×10^{-5}		6.00
80	195.0		1.905×10^{-5}		5.70
85	185.1		8.337×10^{-6}		5.41
90	183.8		3.396×10^{-6}		5.38
95	190.3		1.343×10^{-6}		5.74
100	203.5	3.0×10^{-7}	5.297×10^{-7}	7.4×10^{-7}	6.15
110	265.5	6.0×10^{-8}	9.661×10^{-8}	3.0×10^{-7}	8.06
120	334.5	1.0×10^{-8}	2.438×10^{-8}	6.0×10^{-8}	11.6
130	445.4	4.5×10^{-9}	8.484×10^{-9}	1.6×10^{-8}	16.1
140	549.0	2.0×10^{-9}	3.845×10^{-9}	6.0×10^{-9}	20.6
150	635.2	1.2×10^{-9}	2.070×10^{-9}	3.5×10^{-9}	24.6
160	703.1	6.5×10^{-10}	1.244×10^{-9}	2.0×10^{-9}	26.3
180	781.2	2.4×10^{-10}	5.464×10^{-10}	9.0×10^{-10}	33.2
200	859.3	1.0×10^{-10}	2.789×10^{-10}	3.2×10^{-10}	38.5
250	940.2	4.0×10^{-11}	7.248×10^{-11}	1.6×10^{-10}	46.9
300	972.8	1.6×10^{-11}	2.418×10^{-11}	8.8×10^{-11}	52.5
350	986.5	2.0×10^{-12}	9.158×10^{-12}	6.0×10^{-11}	56.4
400	992.6	3.7×10^{-13}	3.725×10^{-12}	5.0×10^{-11}	59.4
450	995.7	9.0×10^{-14}	1.585×10^{-12}	3.8×10^{-11}	62.2
500	997.3	1.3×10^{-14}	6.967×10^{-13}	3.0×10^{-11}	65.8
600	1000.0		1.454×10^{-13}		79
700	1000.0		3.614×10^{-14}		109
800	1000.0		1.170×10^{-14}		164
900	1000.0		5.245×10^{-15}		225
1000	1000.0		3.019×10^{-15}		268

Table 2.3 - Upper atmosphere of the Earth (Wertz 1978).

2.5.3 Ignored Torques

In addition to the environmental disturbances listed earlier, there are some sources that, due to their small size at the selected satellite altitude, are ignored in this dissertation.

The gravity of the Moon: because the gravity of the Earth is dominant (it is closer and larger), the gravity of the Moon, and the tidal force created by the Earth-Moon system is ignored.

Solar winds and pressure: the Sun's radiation of particles causes solar winds and pressure. Both the solar winds and pressure generate torques on the satellite.

Satellite generated torques: the satellite generated torques are generated by different sources, such as the deployment of antennas or moving parts in the satellite and instruments on-board which generate residual magnetic fields that interact with the Earth's magnetic field. These torques are either short lived, or have a low magnitude, and can therefore be ignored. Other disturbance torques caused by moving parts in the satellite are also neglected.

2.6 Attitude Determination and Control System

The attitude (*i.e.* the orientation) of a spacecraft can be determined by describing the rotation between a spacecraft-fixed reference frame and a known reference frame. This description is accomplished by finding rotations between measured vectors and known quantities.

Regarding the attitude sensors, there are many types of sensors available for observing the attitude and angular rates of a satellite. Common attitude sensors include Sun sensors, Earth sensors, magnetometers, star trackers, GPS and gyroscopes, and are described in Wertz (1978) and Sidi (1997). Gyroscopes are usually used also to measure the angular velocities of the spacecraft.

Some form of attitude control is required to change the attitude of a spacecraft or keep it in a stable position. Three main types of attitude control are usually exploited in aerospace applications. These types of control include spin stabilization, three-axis control techniques, and passive control (Wertz 1978; Sidi 1997; Wertz and Larson 1999). In this thesis an active three-axis control method is considered.

Regarding the actuators for active attitude control, there are basically three main categories of actuators available to actively control a satellite. These are thrusters, wheels and magnetic coils. Three-axis control and stabilization wheels include reaction wheels, momentum wheels, and control moment gyros. Reactions wheels have a variable speed and a continuous and smooth control. Momentum wheels have a nonzero speed and provide angular momentum to the spacecraft. Momentum and reaction wheels are used in many spacecraft to allow an accurate pointing. Control moment gyros are fixed-speed gimballed wheels, but are not often used on small satellites because of their large weight.

The Attitude Determination and Control System (ADCS) can be viewed as two separate subsystems: an Attitude Control System (ACS) (*i.e.* the control actuators and relative actuator sensors) and an Attitude Determination System (ADS) (*i.e.* the attitude and angular velocity sensors).

The presence of four reaction wheels is considered in the ACS. The four reaction wheels are arranged in a tetrahedral configuration. This configuration allows to have an overall three-axis zero momentum bias even if the wheels run with a momentum bias. Moreover, the presence of a fourth redundant actuator allows to maintain the complete attitude controllability also in case of failure of one actuator. Finally, also the presence of flywheel spin rate sensors is assumed and the angular velocities of the actuator flywheels are assumed to be available also for diagnosis purpose.

Both the control torques actuated by the reaction wheels and the satellite attitude and angular velocity measurements and flywheel spin rate measurements can be affected by the occurrence of possible faults on the actuators and sensors.

The sensors assumed to be present on-board in the ADS are two star trackers, each providing a distinct attitude measurement (*i.e.*, two distinct unit quaternions), and three distinct rate gyros, each of them providing a measurement of spacecraft angular rate about a specific body axis. The double hardware redundancy of the attitude sensors is exploited in the FDI scheme to correctly isolate an occurred fault affecting any of the attitude or angular rate sensors. It is worth observing that this double hardware redundancy anyway would be insufficient to correctly isolate a faulty attitude sensor directly by comparing the different attitude measurements by means of a voting scheme, but only to detect the occurrence of a possible fault. Therefore, in this thesis, the isolation of all faults is carried out by means of residual generators exploiting analytical redundancies.

Multiple types of attitude determination sensors could be exploited, also in the context of sensor fusion, by combining different sensory data from several sources and resulting in information, which is more accurate, complete and dependable than it would be possible when these sources were used individually. However, in this thesis only star trackers and rate gyros are considered, measuring the spacecraft attitude and angular velocities respectively.

2.6.1 Reaction Wheel

Reaction wheels are effective active control elements. They are particularly good for variable spin rate control. Active control of spacecraft by using reaction wheels is a fast, flexible, precise way of attitude control and stabilization. On the other hand, it requires rapidly moving parts which implies problems of support and friction. In this dissertation, any effect of viscous and Coulomb friction is neglected and the reaction wheels are assumed to be ideal actuators.

A reaction wheel is an electromechanical device composed of an electric motor that drives a rotor with significant inertia for the mass it has (*i.e.* a flywheel). The wheel can be accelerated (or decelerated) by the motor, which can be integrated in the wheel structure, and this acceleration is produced by a motor torque on the actuator wheel

$$T_w = I_w \dot{\omega}_w$$

where T_w is the torque from the stator to the rotor of the wheel, I_w is the wheel inertia and $\dot{\omega}_w$ is the angular acceleration produced by the motor. The torque on the wheel will generate a reactive torque, defined by Eq. (2.41), with opposite sign on the satellite body, which can be used to control the angular velocity of the satellite:

$$T_{ctrl} = -T_w = -I_w \dot{\omega}_w \tag{2.41}$$

The applied motor torque is the same as the time derivative of the angular moment h_w of the wheel. Hence, for each reaction wheel, the following dynamic equation can be written:

$$\dot{h}_w = I_w \dot{\omega}_w = T_w = -T_{ctrl} \tag{2.42}$$

and in a rotating body frame the total torque produced by a reaction wheel results to be

$$T_{w,tot}^b = \left(\frac{dh_w}{dt} \right)^b + \omega_{ib}^b \times h_w$$

This total torque includes also the gyroscopic effects of the coupling interaction between a spinning flywheel and the rotating satellite. Recalling the general Euler's moment equation (Wertz 1978; Sidi 1997) defined as

$$\dot{h}_{tot} = T_{tot} - \omega \times h_{tot}$$

where $h_{tot} = \sum_{i=1}^{N} I_i \omega_i$ is the summation over all (N) parts within the satellite, the dynamic equations of Eq. (2.22) have to be written by considering also the contributions of the reaction wheels to the total angular momentum. Hence, with $\mathbf{h}_w^b = \begin{bmatrix} h_{w,x} & h_{w,y} & h_{w,z} \end{bmatrix}^T$ representing the total angular momentum of the actuators seen in the satellite body coordinate system, the dynamic equations of Eq. (2.26) are rewritten as

$$\mathbf{I}_{sat}\dot{\boldsymbol{\omega}}_{ib}^b = -\boldsymbol{\omega}_{ib}^b \times (\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^b + \mathbf{h}_w) + \mathbf{T}_{ctrl}^b + \mathbf{T}_{ext}^b \tag{2.43}$$

or, with a simpler notation

$$\mathbf{I}_{sat}\dot{\boldsymbol{\omega}}_{ib}^b = -S(\boldsymbol{\omega}_{ib}^b)(\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^b + \mathbf{h}_w) + \mathbf{T}_{ctrl}^b + \mathbf{T}_{ext}^b \tag{2.44}$$

where the cross product is written as a matrix operation using the skew-symmetric matrix

$$S(\boldsymbol{\omega}_{ib}^b) = \begin{bmatrix} 0 & -\omega_{z,i} & \omega_{y,i} \\ \omega_{z,i} & 0 & -\omega_{x,i} \\ -\omega_{y,i} & \omega_{x,i} & 0 \end{bmatrix}$$

and the control torques about the centre of mass of the satellite actuated by reaction wheels and seen in the satellite body coordinate system are given as

$$\mathbf{T}_{ctrl}^b = -\dot{\mathbf{h}}_w^b \tag{2.45}$$

which gives the rate of change of the total angular momentum from the wheels. Solving the Eq. (2.43) for $\dot{\boldsymbol{\omega}}_{ib}^b$ leads to

$$\dot{\boldsymbol{\omega}}_{ib}^b = -\mathbf{I}_{sat}^{-1} S(\boldsymbol{\omega}_{ib}^b)(\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^b + \mathbf{h}_w^b) + \mathbf{I}_{sat}^{-1}(\mathbf{T}_{ctrl}^b + \mathbf{T}_{ext}^b) = \begin{bmatrix} -\dfrac{(I_{zz}-I_{yy})}{I_{xx}}\omega_{y,i}\omega_{z,i} + \dfrac{(\omega_{z,i}h_{w,y} - \omega_{y,i}h_{w,z})}{I_{xx}} + \dfrac{(T_{ctrl,x}^b + T_{ext,x}^b)}{I_{xx}} \\ -\dfrac{(I_{xx}-I_{zz})}{I_{yy}}\omega_{x,i}\omega_{z,i} + \dfrac{(\omega_{x,i}h_{w,z} - \omega_{z,i}h_{w,x})}{I_{yy}} + \dfrac{(T_{ctrl,y}^b + T_{ext,y}^b)}{I_{yy}} \\ -\dfrac{(I_{yy}-I_{xx})}{I_{zz}}\omega_{x,i}\omega_{y,i} + \dfrac{(\omega_{y,i}h_{w,x} - \omega_{x,i}h_{w,y})}{I_{zz}} + \dfrac{(T_{ctrl,z}^b + T_{ext,z}^b)}{I_{zz}} \end{bmatrix} \tag{2.46}$$

Moreover, since in general the spinning axes of a reaction wheel could be not aligned with the satellite body coordinate system axes, the corresponding control torque of Eq. (2.45) seen in the satellite body frame must be defined by considering the actual (fixed) orientation of the actuator.

The angular momentum vector of a reaction wheel, seen in the satellite body frame, is then needed, using a rotation from wheel coordinates to satellite body coordinates. The components of this angular momentum in the satellite body coordinate system correspond to the projection in the satellite body coordinate system of the angular momentum of the wheel along its axis of rotation. The projection is a column vector defined as

$$h_w^b = \begin{bmatrix} h_{w,x} \\ h_{w,x} \\ h_{w,x} \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} h_w$$

If the wheel assembly consists of several wheels, the directional cosines of the projections of each wheel are the columns of the configuration (direction) matrix for the wheels, such that

$$\mathbf{h}_w^b = \begin{bmatrix} h_{w,x}^b \\ h_{w,y}^b \\ h_{w,z}^b \end{bmatrix} = \begin{bmatrix} a_{x,1} & a_{x,2} & \cdots & a_{x,N} \\ a_{y,1} & a_{y,2} & \cdots & a_{y,N} \\ a_{z,1} & a_{z,2} & \cdots & a_{z,N} \end{bmatrix} \begin{bmatrix} h_{w,1} \\ h_{w,2} \\ \vdots \\ h_{w,N} \end{bmatrix}$$

Abbreviated to vector form, this is

$$\mathbf{h}_w^b = \mathbf{A}_w \mathbf{h}_w \tag{2.47}$$

The matrix $\mathbf{A}_w$ has three rows and a number of columns equal to the number of reaction wheels in the satellite, in this case equal to four. The four reaction wheels are arranged in a tetrahedral configuration, as shown in Fig. 2.5.

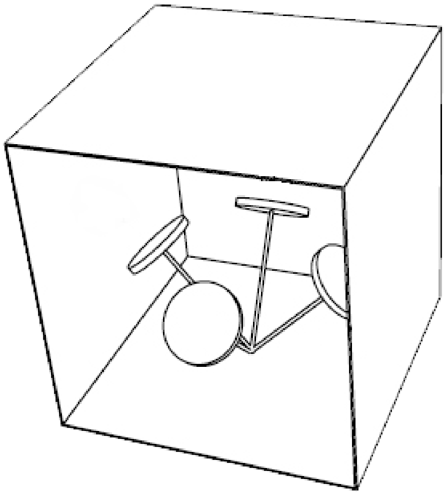


Figure 2.5 – Tetrahedral configuration of the satellite reaction wheels.

This configuration allows having an overall three-axis zero momentum bias even if the wheels run with a momentum bias. Moreover, the presence of a fourth redundant actuator allows maintaining the complete attitude controllability on the three axes also in case of failure of one actuator. The configuration matrix of the actuators considered in this thesis is given by

$$\mathbf{A}_w = \begin{bmatrix} \sqrt{1/3} & \sqrt{1/3} & -\sqrt{1/3} & -\sqrt{1/3} \\ \sqrt{2/3} & -\sqrt{2/3} & 0 & 0 \\ 0 & 0 & -\sqrt{2/3} & \sqrt{2/3} \end{bmatrix} \tag{2.48}$$

Therefore, the overall control torques about the centre of mass of the satellite actuated by the four considered reaction wheels in the selected configuration and seen in the satellite body coordinate system are defined as

$$\mathbf{T}_{ctrl}^{b}=\begin{bmatrix}T_{ctrl,x}^{b}\\T_{ctrl,y}^{b}\\T_{ctrl,z}^{b}\end{bmatrix}=\begin{bmatrix}\sqrt{1/3}T_{ctrl,1}+\sqrt{1/3}T_{ctrl,2}-\sqrt{1/3}T_{ctrl,3}-\sqrt{1/3}T_{ctrl,4}\\\sqrt{2/3}T_{ctrl,1}-\sqrt{2/3}T_{ctrl,2}\\-\sqrt{2/3}T_{ctrl,3}+\sqrt{2/3}T_{ctrl,4}\end{bmatrix}=\mathbf{A}_{w}\begin{bmatrix}T_{ctrl,1}\\T_{ctrl,2}\\T_{ctrl,3}\\T_{ctrl,4}\end{bmatrix}=\mathbf{A}_{w}\mathbf{T}_{ctrl} \quad (2.49)$$

Finally, since sensors providing measurements of the spin rates of the four flywheels are assumed to be present in the actuator subsystems, thus the values of $\boldsymbol{\omega}_w=\begin{bmatrix}\omega_{w,1} & \omega_{w,2} & \omega_{w,3} & \omega_{w,4}\end{bmatrix}^T$ and of the angular momentums $\mathbf{h}_w=\mathbf{I}_w\boldsymbol{\omega}_w=\begin{bmatrix}h_{w,1} & h_{w,2} & h_{w,3} & h_{w,4}\end{bmatrix}^T$ are available for diagnosis purpose, with the moment of inertia I_w of each wheel known. These actuator sensors could consist, for example, in incremental (relative) optical encoders mounted on the spin axes of the reaction wheels.

Optical encoders consist of a spinning disc made of glass or plastic with transparent and opaque areas. This disk is connected with the spinning shaft or wheel. An absolute optical encoder has multiple code rings with various binary weightings which provide a data word representing the absolute position of the encoder within one revolution. A light source and photo detector array reads the optical pattern that results from the disc position at any one time.

Differently, an incremental encoder works by providing two pulse outputs, called quadrature outputs due to their wave forms being 90 degrees out of phase, which provide no usable position information in their own right. Rather, an incremental rotary encoder provides cyclical outputs (only) when the encoder is rotated. It reports an incremental change in position of the encoder to an external counting electronics performing a pulse counting.

The pulse signals are decoded to produce a count up pulse or a count down pulse. The rate of rotation can be measured by determining the time between pulses provided by the encoder. Hence, the output of incremental encoders provides information about the motion of the shaft, which is typically further processed by a controlling device, such as a microprocessor or microcontroller, into information such as speed, sense of direction and also position. Rotary encoders with a single output (*i.e.* pulsers) cannot be used to sense direction of motion.

The sensor measurement can be modelled as the true flywheel spin rate with added noise. The measured spin rate $\boldsymbol{\omega}_{wheel}$ can be modelled as

$$\boldsymbol{\omega}_{wheel}=\boldsymbol{\omega}_{w}+\boldsymbol{\omega}_{w,n} \quad (2.50)$$

where $\boldsymbol{\omega}_w$ is the actual flywheel spin rate, and $\boldsymbol{\omega}_{w,n}$ is the additive measurement noise, modelled in this thesis by a Gaussian white noise with variance

$$E\left[\omega_{w,n,i}^{2}\right]=\sigma_{\omega_{w,n,i}}^{2} \quad \text{for } i=1,2,3,4$$

2.6.2 Star Tracker

A star tracker is a light sensitive precision instrument, which determines the attitude of a satellite by observing stars with high precision and comparing with a library of constellations to determine the attitude of a spacecraft. The two major elements in a star tracker are a digital camera and processing unit. The star tracker observes the stars with the camera, and compares these observations (*i.e.* star light intensity and the relative positions) with an on-board star catalogue. This gives the angles between the observed star and a reference frame in the satellite. If only one star (or clusters of stars) is observed, this will only give an accurate information about the attitude in two dimensions. By observing at least two different remote stars (or cluster of stars) the star tracker can determine a well-defined three-axis attitude representation in the inertial reference frame. Star trackers are

accurate and work at any point in an orbit. The accuracy of the star tracker depends on the number of stars observed, the star catalogue used, the quality of the optic and the resolution in the digital camera. A typical star sensor used in space applications has an accuracy of a few arcseconds (fine pointing information). However, high accuracy star trackers are often massive, complex, and expensive. Searching the library of constellations is time-consuming, so for high accuracy and rapid response, star trackers are often used along with gyroscopes.
The attitude sensor measurement can be modelled as the true attitude with added noise. Because of the required unity of the measured quaternion $\bar{\mathbf{q}}_{star}$, a sensor model with additive noise is not directly applicable, and the star sensor can be modelled as

$$\bar{\mathbf{q}}_{star} = \bar{\mathbf{q}} \otimes \bar{\mathbf{q}}_n \tag{2.51}$$

where $\bar{\mathbf{q}}$ is the actual attitude, and $\bar{\mathbf{q}}_n$ is the star sensor measurement noise, represented as Euler parameters. Let's consider, for example, a generic attitude sensor with an assumed accuracy of 1 arcsecond around all axis. Since the sensor model produces the attitude as a quaternion, it is convenient to represent the accuracy in the magnitude of the quaternion. By regarding the individual axis angles accuracies as an composite rotation, the accuracies may be represented as the unit quaternion $\bar{\mathbf{q}}_a$ as

$$\Theta_a = \begin{bmatrix} \phi_a \\ \theta_a \\ \psi_a \end{bmatrix} = \begin{bmatrix} 0.000278 \\ 0.000278 \\ 0.000278 \end{bmatrix} \deg \Rightarrow q_a \approx \begin{bmatrix} 2.4 \cdot 10^{-6} \\ 2.4 \cdot 10^{-6} \\ 2.4 \cdot 10^{-6} \\ 1 \end{bmatrix} \tag{2.52}$$

By using Eq. (2.52), the star sensor measurement noise can be modelled as

$$\bar{\mathbf{q}}_n = \begin{bmatrix} \mathbf{q}_n \\ \sqrt{1 - \|\mathbf{q}_n\|^2} \end{bmatrix}$$

where the vector part $\mathbf{q}_n$ of the noise quaternion is modelled in this thesis by Gaussian white noise with variance

$$E\left[\mathbf{q}_{n,\mathrm{i}}^2\right] = \sigma_{\mathbf{q}_{n,i}}^2 \quad \text{for } i = 1,2,3$$
$$\sigma_{\mathbf{q}_{n,i}} = 2.4 \cdot 10^{-6}$$

However, when using the measurement in attitude determination, it could be convenient to model the measurement noise as additive. Hence, the additive measurement noise $\bar{\mathbf{v}}_{star}$ can be represented as

$$\bar{\mathbf{v}}_{star} = \bar{\mathbf{q}} \otimes \bar{\mathbf{q}}_n - \bar{\mathbf{q}} \tag{2.53}$$

It can be seen that, with the formulation of the attitude sensor model of Eq. (2.53), the measurement noise becomes scaled. The noise model and noise covariance can be represented on component form as

$$\bar{\mathbf{v}}_{star} = \begin{bmatrix} q_4 q_{4n} - \mathbf{q}^T \mathbf{q}_n - q_4 \\ q_4 \mathbf{q} + q_{4n} \mathbf{q} + S(\mathbf{q}) \mathbf{q}_n - \mathbf{q} \end{bmatrix}$$

$$E\left[\bar{\mathbf{v}}_{star}^2\right] = \sigma_{\bar{\mathbf{v}}_{star}}^2 = \begin{bmatrix} q_4 E\left[q_{4n}^2\right] - \mathbf{q}^T E\left[\mathbf{q}_n^2\right] - q_4 \\ q_4 E\left[\mathbf{q}_n^2\right] + E\left[q_{4n}^2\right] \mathbf{q} + S(\mathbf{q}) E\left[\mathbf{q}_n^2\right] - \mathbf{q} \end{bmatrix}$$

Concerning the attitude sensor faults, it is worth noting that, since each attitude sensor provides a complete quaternion vector measurement, in case of attitude sensor fault all the components of the measured quaternion vector could present differences from their fault-free values depending on the fault characteristics and actual spacecraft attitude. For this reason, it is not possible to consider a possible fault affecting an attitude sensor as a scalar fault term affecting a single quaternion vector component. In general, in case of attitude sensor fault, all the quaternion vector components can be affected by the occurred sensor fault.

In fact, similarly to the model of the noisy quaternion vector described above, the attitude sensor fault can be modelled as the true attitude quaternion vector with additional fault terms. Because of the required unity of the measured quaternion $\bar{\mathbf{q}}_{star}$, a sensor model with additive quaternion fault is not directly applicable, and the faulty output of the star sensor can be modelled as

$$\bar{\mathbf{q}}_{star} = \bar{\mathbf{q}}_{true} \otimes \bar{\mathbf{q}}_f \tag{2.54}$$

where $\bar{\mathbf{q}}_{true}$ is the actual attitude, and $\bar{\mathbf{q}}_f$ is the sensor fault vector, represented as Euler parameters. Hence, the equivalent additive physical fault $\bar{\mathbf{F}}_{\bar{\mathbf{q}}_{star}}$ can be represented as

$$\bar{\mathbf{F}}_{\bar{\mathbf{q}}_{star}} = \bar{\mathbf{q}}_{star} \otimes \bar{\mathbf{q}}_f - \bar{\mathbf{q}}_{star} \tag{2.55}$$

Therefore, considering both the presence of the measurement noise and sensor fault, the overall value of the measured quaternion is given by

$$\bar{\mathbf{q}}_{star_{measured}} = (\bar{\mathbf{q}}_{true} \otimes \bar{\mathbf{q}}_f) \otimes \bar{\mathbf{q}}_n \tag{2.56}$$

As already stated, since in this thesis a LEO Earth-pointing satellite is considered, it could be convenient to define the satellite attitude as the rotation between the body-fixed reference frame and the orbital reference frame. Moreover, Eqs. (2.38) and (2.40) of the external disturbance torques explicitly exploit the spacecraft attitude values with respect to the orbital reference frame.

However, it is worth noting that star trackers actually provide attitude measurements with respect to the inertial reference frame of the fixed stars. Hence, a relation to convert these inertial measurements into corresponding attitude measurements with respect to the orbital reference frame is needed. Obviously, given an inertial spacecraft attitude, the corresponding orbital attitude depends also on the instantaneous orientation of the LVLH orbital reference frame in the inertial reference frame. If the satellite position along the orbit path is assumed to be known, it can be used to derive the orientation of the orbital reference frame with respect to the inertial reference frame, and consequently the spacecraft attitude with respect to the orbital frame.

Considering an equatorial circular prograde (counter-clockwise) orbit, the satellite position along the orbit path can be derived simply by integrating the angular velocity ω_o of the satellite orbit with an assumed known initial orientation condition of the orbital reference frame with respect to the inertial one. At the initial time of the simulation, the orbital reference frame is assumed to have the x_o-axis aligned with the inertial x_i-axis, *i.e.* pointing in the vernal equinox direction. The z_o-axis is assumed to be aligned with the inertial y_i-axis, and the y_o-axis completing the right-hand system is assumed to be pointing in the opposite direction of the inertial z_i-axis, *i.e.* in the direction antinormal to the orbital plane in the inertial reference frame. Hence, the orbital frame results to be initially rotated of $\phi_o = -\pi/2$ rad about the x_i-axis. Therefore, the orbital frame orientation at each instant of simulation can be defined as

$$\bar{\mathbf{q}}_{\Omega_0} = \begin{bmatrix} \frac{\sqrt{2}}{2}\cos\frac{-\Omega_o}{2} \\ -\frac{\sqrt{2}}{2}\cos\frac{-\Omega_o}{2} \\ \frac{\sqrt{2}}{2}\sin\frac{-\Omega_o}{2} \\ -\frac{\sqrt{2}}{2}\sin\frac{-\Omega_o}{2} \end{bmatrix}$$

representing the combination of two consecutive rotations $\phi_o = -\pi/2$ rad about the x_i-axis and $-\Omega_o$ rad about the rotated y_o-axis, where the orbit position of the satellite in the inertial reference frame is defined by a rotation angle about the z_i-axis given by

$$\Omega_o = \int_t \omega_o dt \qquad \Omega_o(0) = 0$$

and, given attitude values with respect to the inertial frame, the spacecraft attitude with respect to the orbital frame can be derived by the relation

$$\bar{\mathbf{q}}_o = \bar{\mathbf{q}}_{\Omega_o} \otimes \bar{\mathbf{q}}_i = \begin{bmatrix} q_{4,i} & q_{3,i} & -q_{2,i} & q_{1,i} \\ -q_{3,i} & q_{4,i} & q_{1,i} & q_{2,i} \\ q_{2,i} & -q_{1,i} & q_{4,i} & q_{3,i} \\ -q_{1,i} & -q_{2,i} & -q_{3,i} & q_{4,i} \end{bmatrix} \bar{\mathbf{q}}_{\Omega_o} \tag{2.57}$$

The dynamic and kinematic equations actually exploited for the design of the FDI/FDD systems are given by Eqs. (2.46) and (2.30).

2.6.3 Rate Gyro

A rate gyro is a type of gyroscope, which rather than indicating direction, indicates the rate of change of angle with time. If a gyro has only one gimbal ring, with consequently only one plane of freedom, it can be adapted for use as a rate gyro to measure a rate of angular movement.

In general, a gyroscope is a device for measuring or maintaining orientation, based on the principles of angular momentum. Mechanical gyroscopes typically comprise a spinning wheel or disc in which the axle is free to assume any orientation.

When mounted in (outer and inner) gimbals, which minimize external torque, the orientation of the spin axis remains nearly fixed, regardless of the mounting platform's motion. Therefore, they can be used to sense or measure the pitch, roll and yaw attitude angles in a spacecraft or aircraft.

Gyroscopes based on other operating principles also exist, such as vibrating structure gyroscopes (including MEMS gyros), solid-state ring lasers, fiber optic gyroscopes, and the extremely sensitive quantum gyroscopes. These types of gyros operate on the principle of the Sagnac effect based on the interference of light which has passed through a coil of optical fiber or counter-propagating laser beams.

Rate gyros are used to measure the rates of rotation of the spacecraft around the sensor axes. If the sensor is aligned with a selected body-fixed frame, it provides measurements of the satellite angular rate around the body axis with respect the inertial reference frame.

Gyroscopes can also determine the attitude by measuring and integrating the rate of rotation of the spacecraft. They are located internal to the spacecraft and work at all points in an orbit. Gyroscopes have a high accuracy for limited time intervals. Some disadvantages exist with gyroscopes. Since they measure a change instead of absolute attitude, gyroscopes must be used along with other attitude hardware to obtain full attitude measurements. They also are subject to drift, and since they have moving parts, there is more complexity. In this thesis, it is assumed that the three considered rate gyros provide only angular velocity information and not attitude information.

In this case, the sensor measurement can be modelled as the true angular velocity with added noise. The measured angular velocity $\boldsymbol{\omega}_{gyro}$ can be modelled as

$$\boldsymbol{\omega}_{gyro} = \boldsymbol{\omega} + \boldsymbol{\omega}_n \tag{2.58}$$

where $\boldsymbol{\omega}$ is the actual angular velocity, and $\boldsymbol{\omega}_n$ is the additive measurement noise, modelled in this thesis by a Gaussian white noise with variance

$$E\left[\omega_{n,i}^2\right] = \sigma_{\omega_{n,i}}^2 \quad \text{for } i = 1,2,3$$

A sensor model with additive fault is also directly applicable for this sensor, and the faulty output of the rate gyro sensor can be modelled as

$$\boldsymbol{\omega}_{gyro} = \boldsymbol{\omega}_{true} + \boldsymbol{\omega}_f \tag{2.59}$$

where $\boldsymbol{\omega}_{true}$ is the actual attitude, and $\boldsymbol{\omega}_f = \overline{\mathbf{F}}_{\boldsymbol{\omega}_{gyro}}$ is the sensor fault vector. Therefore, considering both the presence of the measurement noise and sensor fault, the overall value of the measured angular velocity vector is given by

$$\boldsymbol{\omega}_{gyro_{measured}} = \boldsymbol{\omega}_{true} + \boldsymbol{\omega}_f + \boldsymbol{\omega}_n \tag{2.60}$$

3 MODEL-BASED FAULT DETECTION AND ISOLATION

This chapter provides an overview of the FDI problem, together with the general description of the residual generation method and fault detection and isolation scheme. An exploited method for sensor and actuator fault modelling and the NLGA for affine nonlinear systems, which is exploited for disturbance decoupling and fault detection, isolation and estimation, are illustrated as well. Finally, an incremental procedure for the design of the residual filters is described considering to have modelled all the fault inputs in an affine form. This chapter aims to focus mainly on the FDI problem. However, many of the described aspects are in common with the FDD problem and the exploited methods can be used also for FDD.

3.1 General Overview

All the terminologies defined in this dissertation are based on information from the International Federation Automatic Control (IFAC) SAFEPROCESS Technical Committee and associate the updated literatures listed herein (Isermann and Ballé 1997; Chen and Patton 1999; Isermann 2006).
A *fault* is defined as an unexpected change of system function or, with an alternative definition, as a non permitted deviation of a characteristic property from the acceptable/usual/standard condition, which leads to the inability of the system to fulfil the intended purpose (Isermann 1984). However, a fault may not represent a physical failure or breakdown. Such a fault or malfunction prevents or disturbs the normal operation of an automatic system, thus causing an unacceptable degradation of the performance of the system or even leading to dangerous situations. Usually, the term *fault* denotes a malfunction, which may be tolerable at its present stage, whereas the term *failure* suggests a complete breakdown of a system component or function, *i.e.* a permanent interruption of a system's ability to perform a required function under specified operating conditions. A fault must be diagnosed as early as possible even if it is tolerable at its early stage, in order to prevent any serious consequences.
A monitoring system which is used to detect faults and diagnose their location and significance in a system is called a *fault diagnosis system* (Frank 1990; Chen and Patton 1999). Such a system normally performs the following tasks:

1. **Fault detection:** a binary decision is made, either that something has gone wrong or that everything is fine;
2. **Fault isolation:** the location of the fault is determined, *i.e.* which sensor, actuator or system component has become faulty;
3. **Fault identification and fault estimation:** the size and type of the fault is estimated.

Fault detection represents the most important task for any practical system and isolation is almost equally important. On the other hand, fault identification may not be essential, whilst undoubtedly helpful, if no control reconfiguration and fault accommodation actions are involved. Hence, a fault diagnosis system is usually built to perform one or more functionalities as shown in Fig. 3.1.

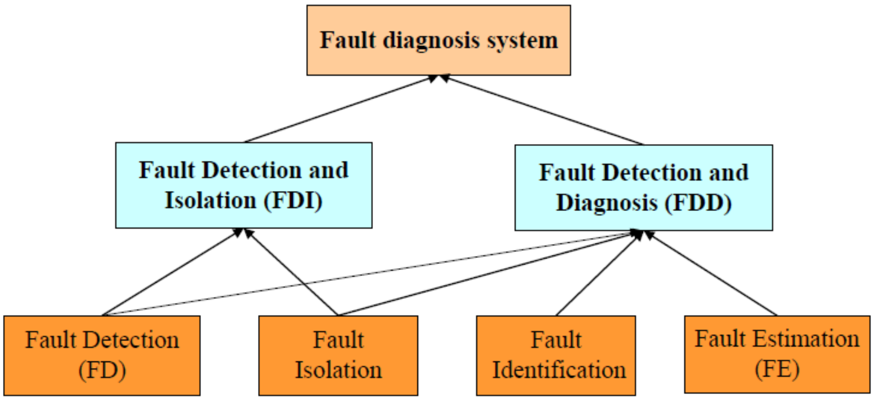


Figure 3.1 - Topological illustration of fault diagnosis functional relationships (Sun 2013).

The difference between the roles of Fault Detection and Isolation (FDI) and Fault Detection and Diagnosis (FDD) must be clear in order to avoid any confusion, since the terms isolation and identification share the same initials but correspond to different functional cases. FDI aims to locate and isolate the faulty components in the system. FDD, however, is intended to know the detailed attributes of detected faults, *e.g.* faults severity and the fault identification, *i.e.* the determination of fault location and strength, is required in FDD. Fault Estimation (FE) is functionally similar as Fault Identification (FI) but focuses on reconstruction of the fault signal using estimation-based methods (Ding 2013). In this chapter, the Fault Detection and Isolation (FDI) tasks are discussed, whilst the Fault Estimation (FE) will be explicitly discussed in Chapter 5.

In practice, the most frequently used diagnosis method consists in monitoring the level (or trend) of a particular signal, and taking action when it reaches a given threshold value. However, even if this method of limit checking is simple to implement, it has serious drawbacks. Firstly, there is the possibility of false alarms in the event of noise, input variations and change of operating point. Secondly, a single fault could cause many system signals to exceed their limits and appear as multiple faults, making difficult to perform fault isolation. The use of consistency checking for a number of system signals can eliminate these problems and enhance the detection and isolation or fault diagnosis capability of a generic automated system. However, a mathematical model providing functional relationships among different system signals is required.

In order to evaluate the inconsistency, some form of redundancy is required. A traditional approach to fault diagnosis in the wider application context is based on *hardware* (or physical/parallel) redundancy methods, which exploit multiple lanes of sensors, actuators, computers and software to measure and/or control a particular variable. Usually, a voting scheme is exploited in the hardware redundant system to decide if and when a fault has occurred and its likely location amongst redundant system components. However, the major problem of this approach consists in the required extra equipment, maintenance cost and additional space required to accommodate the hardware. In order to improve the system reliability but limit the cost, *analytical* (functional) redundancy can be exploited, using dissimilar measured values together to cross-check each other, rather than replicating each hardware individually. In this case, redundant analytical (or functional) relationships between various measured variables of the monitored process (*e.g.* inputs/outputs, outputs/outputs, inputs/inputs) are used (Beard 1971). No additional hardware components or sensors are required to realize the FDI algorithm, which can be implemented via software. A comparison between the analytical and hardware redundancy concepts is shown in Fig. 3.2.

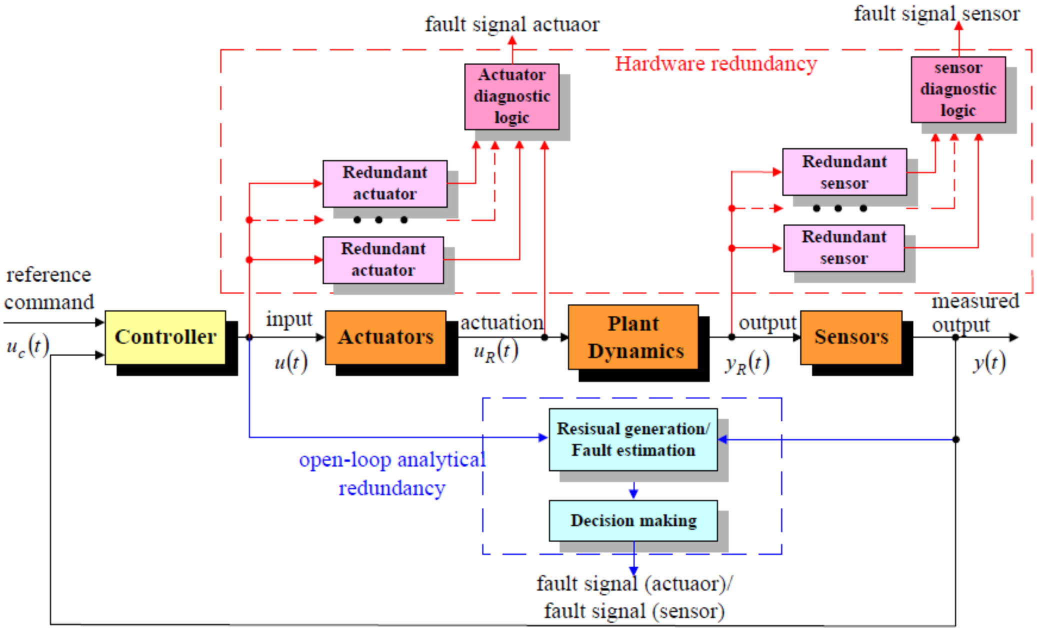


Figure 3.2 - Comparison between FDI hardware redundancy and open-loop analytical (software) redundancy concepts (Sun 2013).

No additional hardware faults are introduced into an analytical redundant scheme, because no extra hardware is required. Hence, analytical redundancy is potentially more reliable than hardware redundancy. In analytical redundancy schemes, the resulting difference generated from the consistency checking of different variables is called as a *residual signal*. This signal should be zero-valued when the system is normal, and should diverge from zero when a fault occurs in the system.

This property of the residual is used in fault diagnosis to determine whether or not faults have occurred. Analytical redundancy exploits a mathematical model of the monitored process, and is therefore referred as *model-based approach*. Normally, consistency checking is achieved through a comparison between a measured signal and its estimation, which is generated by an a priori known mathematical model of the considered system. The residual (fault indicator, diagnostic signal) gives the difference between these two signals, *i.e.* the measured and estimated ones (Frank 1990; Chen and Patton 1999).

In general, many research studies have been devoted to the subject of FDI/FDD in which a mathematical model is used to realise the FDI/FDD function in real-time. A general overview of the FDD problem will be given with more details in Chapter 5. Several design methodologies for model-based FDI/FDD have been studied, which are substantially based on explicit and quantitative methods, *i.e.* observer-based methods, parameter estimation, parity space, as shown in Fig. 3.3 and introduced as follows:

Observer based FDI/FDD: This technique has been developed under the framework of the well-studied advanced control theory. This is regarded as an effective method for designing observers using efficient and reliable algorithms for data processing to reconstruct system variables. Beard (1971) and Clark (1978) started the pioneer work of this approach, and systematic design approach can be found in Chen and Patton (1999) and Ding (2013). In this approach, an observer is used to estimate the actual system outputs. The residuals/FE signals are then constructed via suitable functions of the output estimation error between the measured and estimated outputs. Explicit model-based FDI/FDD approaches are the most preferred and most studied (Willsky 1976; Clark 1978; Isermann and Ballé 1997; Chen and Patton 1999; Patton *et al.* 2000; Isermann 2011; Ding 2013).

Parity equation based FDI/FDD: A straightforward model-based method of FDI is to use an input-output model with fixed structure and run it in parallel with the process, thereby forming an output error. In this approach, the residual signals are generated based upon consistency checking (or parity checking) on system input and output data over a given time window (Chow and Willsky 1984; Gertler 1998). Actually, once the design objectives are determined, the parity equation and observer based designs have some correspondence or equivalence under certain conditions, as residual generation methods (Gertler 1991; Patton and Chen 1991, 1994; Patton 1997a, 1997b; Hwang *et al.* 2010; Ding 2013).

Parameter estimation based FDI/FDD: This approach is investigated in terms of system identification techniques (Isermann 1984, 2011; Isermann and Ballé 1997). The faults are reflected in the physical system parameters and then the idea of the fault detection is based on the comparison between online estimation of system parameters and the parameters of the fault-free reference model. In most practical cases, the process parameters are only partially known or are not known at all. If the basic model structure of the process is known, the process parameters can be determined with parameter estimation methods by measuring input and output signals. Parameters can be identified by non-recursive or recursive methods or numerical optimisation methods. As the system parameters are obtained, fault estimation can be achieved in some degree.

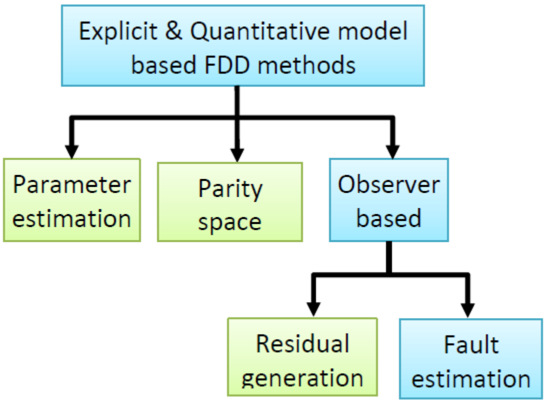


Figure 3.3 - Classification of explicit and quantitative model-based FDI/FDD methods (Shi 2013).

Since model-based FDI makes use of mathematical models of the supervised systems, accurate models are needed. However, a perfectly accurate and complete mathematical model of a physical system is never available. In fact, system parameters may vary with time in an uncertain manner, and characteristics of disturbances and noise are generally unknown, and thus not accurately modelled. Hence, there is always a mismatch between the actual process and its mathematical model even in case of no faults. Such discrepancies cause difficulties in FDI applications, in particular, since they act as sources of possible false alarms and missed alarms potentially corrupting the FDI system performance. The effect of modelling uncertainties, disturbances and noise is therefore the most crucial point in the model–based FDI concept and the solution to this problem is the key for its practical applicability.

To overcome these problems, a model-based FDI scheme has to be made robust, *i.e.* insensitive or even invariant to modelling uncertainty, whilst without reducing the sensitivity to faults. In fact, a simple reduction of the sensitivity to modelling uncertainty sometimes does not solve the problem since the sensitivity reduction may be associated with a reduction of the sensitivity to faults (Frank 1990; Gertler 1998; Chen and Patton 1999). Hence, a more meaningful formulation of the FDI problem is to increase insensitivity to modelling uncertainty in order to provide increasing fault sensitivity. Furthermore, the robustness as well as the sensitivity properties must be independent of the particular fault and disturbance mode. Using proper disturbance decoupling methods, it is possible to design residual signals which are not sensitive to uncertainties. In Section 3.6 it will be shown that, provided certain conditions can be met, complete decoupling of the residuals from disturbances can be achieved, whilst the sensitivity of the residual to faults is maintained. As the faults are also modelled in the form of external signals, this method additionally provides tools for the purpose of fault isolation. Fault isolation requires the decoupling of the effects of different faults on the residual and this, in turn, allows for decisions on which fault or faults out of a given set of possible faults has actually occurred.

Faults are detected by setting a (fixed or variable) threshold on a residual quantity generated from the difference between real measurements and estimates of them obtained using the mathematical model. A number of residual can be designed with each having special sensitivity to individual faults (or limited sets of faults) occurring in different locations in the system. The subsequent analysis of each residual, once the corresponding threshold is exceeded, then leads to fault isolation.

The conceptual structure of a model-based fault diagnosis system consists of two main stages of residual generation and decision making, as firstly suggested in (Chow and Willsky 1980) and shown in Fig. 3.4. These two main stages are described as follows:

1. **Residual generation:** A diagnostic signal (residual) indicating the occurrence of a fault is generated using available input and output information from the monitored system. This auxiliary signal is designed to reflect the onset of a possible fault in the analysed system and the inconsistency between the actual and expected behaviour of the system, and should be normally zero or close to zero when no fault is present. When the fault occurs the residual becomes distinguishably different from zero. This means that the residual is characteristically independent of system inputs and outputs in ideal conditions. The algorithm exploited to generate residuals is called a residual generator. Thus, residual generation is the procedure for extracting fault symptoms from the system, with the fault symptoms represented by the residual signal. The residual should carry only fault information, with a loss of fault information as small as possible;
2. **Decision-making:** The residuals are analysed in order to determine the likelihood of faults, and a decision rule is then applied to determine if any fault is occurred. A decision process may consist of a simple threshold test on the instantaneous values or moving averages of the residuals, or methods of statistical decision theory.

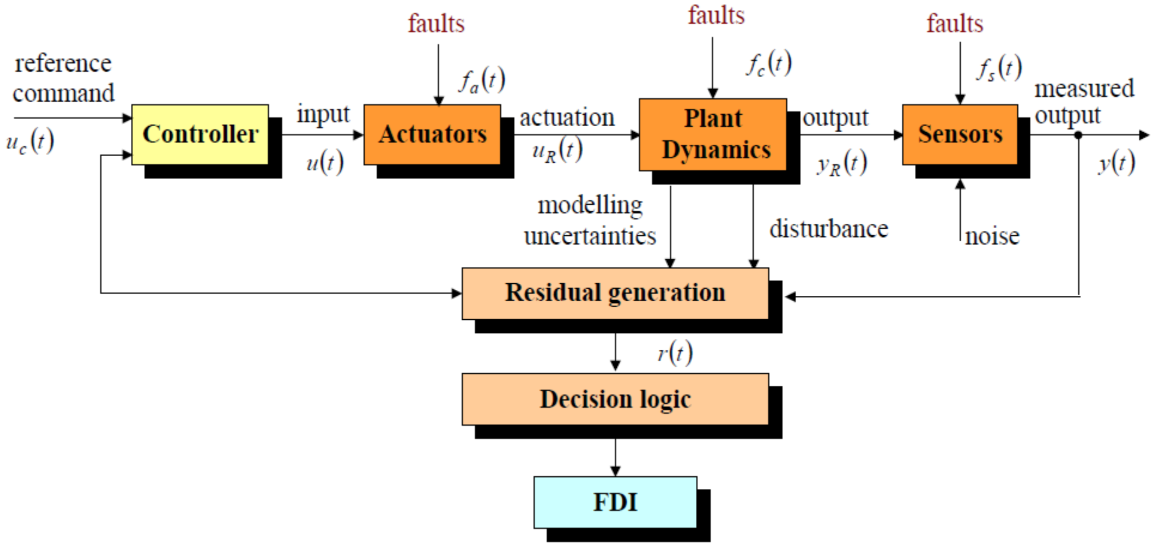


Figure 3.4 - Schematic description of model-based approach to FDI (Sun 2013).

3.2 Residual Generators for Model-based FDI

In practice, the most frequently used FDI approach consists in using information known a priori about the characteristics of certain signals. However, in this way, it is necessary to have a priori information about the characteristics of the signals and it is not possible to avoid dependence of these characteristics on operating states of the system, which are not known a priori and can change beforehand. In order to eliminate the shortcomings of the traditional methods, the modern model-based approaches have introduced residuals, which are independent of the system operating state and respond to faults in characteristic manners. In essence, residuals are quantities that represent the inconsistency between the actual system variables and the mathematical model. Based on the mathematical model, many invariant relations (dynamic or static) among different system variables can be derived, and any violation of these relations can be used as residuals (Chen and Patton 1999).

Therefore, the residual generation can be interpreted in terms of redundant signal structure, as shown in Fig. 3.5. In this structure, the system $F_1(u, y)$ generates an auxiliary (redundant) signal $z(t)$ which, together with $y(t)$ generate the residual $r(t)$ which satisfy the following relations in the fault-free case:

$$\begin{cases} z(t) = F_1(y(t), u(t)) \\ r(t) = F_2(y(t), z(t)) = 0 \end{cases} \tag{3.1}$$

for the fault-free case. When any fault occurs in system (fault-case), this invariant relation is violated and the residual diverge from zero. The simplest approach to residual generation is the use of system duplication, *i.e.* the system $z(t) = F_1(u)$ is made identical to the original system model and has the same input signal as the system. In this case, the signal y would not be required in the system block $F_1(u)$, which is then simply a system simulator (*simulator-based FDI*). The signal z is the simulated output of the system, and the residual is the difference between z and y. The advantage of this method is its simplicity but, as a disadvantage, the stability of the simulator cannot be guaranteed when the system being monitored is unstable, as a consequence of the open-loop system model in FDI (although it is under feedback control).

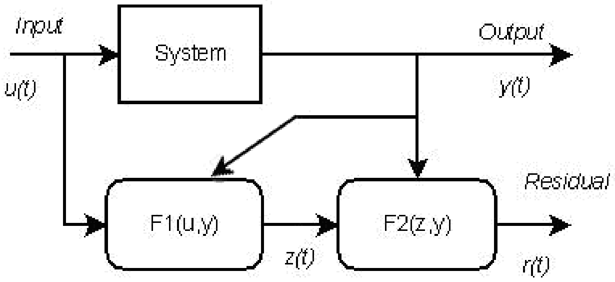


Figure 3.5 - Redundant signal structure in residual generation.

A direct extension to the simulator-based residual generation consists in replacing the simulator by an asymptotically stable output estimator (*observer-based FDI*), which requires both system input and output. In this case, the system $F_1(u, y)$ uses both signal u and y to generate an estimation of the output, and the system $F_2(z, y)$ can be defined as $F_2(z, y) = K(y - z)$ with K as a static (or dynamic) weighting matrix of the estimation errors.

From Fig. 3.6 it can be seen that the system model required in model-based FDI is the open-loop system model although the considered system is in the control loop. This is because the input and output information required in model-based FDI is related to the open-loop system. Hence, it is not necessary to consider the controller in the design of a fault diagnosis scheme.

This is consistent with the separation principle in control theory because fault diagnosis can be broadly treated as an observation problem. Once the input to the actuators is available, the fault diagnosis problem is the same no matter how the system is working in open-loop or in closed-loop. When the actual input $u(t)$ to the actuator is not available, the reference command $u_c(t)$ is used in FDI. Hence, the model involved is the relationship between the reference command $u_c(t)$ and the measured output $y(t)$, *i.e.* the closed-loop model. In such cases, the controller plays an important role in the design of diagnostic schemes. A robust controller may desensitize fault effects and make the diagnosis very difficult.

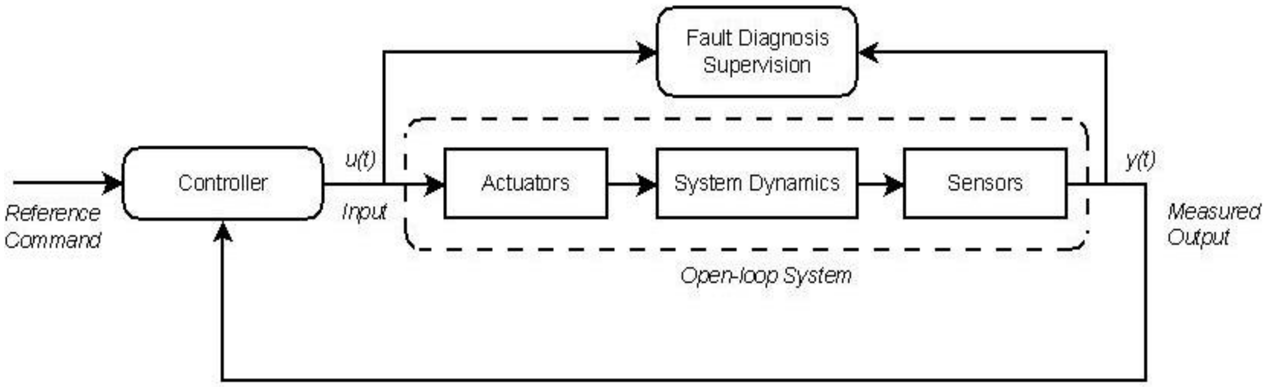


Figure 3.6 - Fault diagnosis and control loop.

A general fault $f(t)$ can be detected by comparing the residual evaluation function $J(t(t))$ with a (fixed or variable) threshold function $T(t)$ according to the following test (Chen and Patton 1999):

$$\begin{cases} J(r(t)) \le T(t) & \text{for} \quad f(t) = 0 \\ J(r(t)) > T(t) & \text{for} \quad f(t) \ne 0 \end{cases} \tag{3.2}$$

If the test is positive (*i.e.* the threshold is exceeded by the residual evaluation function) it can be hypothesized that a fault is likely. This test works especially well with fixed thresholds T if the process operates approximately in steady-state and it reacts after relatively large feature, *i.e.* after either a large sudden or a long-lasting gradually increasing fault. These thresholds can be defined in many ways, on the basis of the chosen residual evaluation functions. In practice, if the residual signal is represented by a stochastic variable $r(t)$, *e.g.* due to the presence of measurement noise, mean value and variance are computed as follows:

$$\bar{r} = E\left[r(t)\right] = \frac{1}{N}\sum_{t=1}^{N} r(t)$$

$$\sigma_r^2 = E\left[(r(t) - \bar{r})^2\right] = \frac{1}{N}\sum_{t=1}^{N} (r(t) - \bar{r})^2$$

where $\bar{r}$ and σ_r^2 are the normal values for the mean and variance of the fault–free residual, respectively. N is the number of samples of the vector $r(t)$. Therefore, the threshold test for FDI of Eq. (3.2) is rewritten as

$$\begin{array}{ll} \bar{r} - \nu\sigma_r \le r(t) \le \bar{r} + \nu\sigma_r & \text{for} \quad f(t) = 0 \\ r(t) < \bar{r} - \nu\sigma_r \quad \text{or} \quad r(t) > \bar{r} + \nu\sigma_r & \text{for} \quad f(t) = 0 \end{array} \tag{3.3}$$

i.e. the comparison of $r(t)$ with respect to its statistical normal values. In order to separate normal from faulty behaviour, the tolerance parameter ν (normally $\nu \ge 3$) is selected and properly tuned. Hence, by a proper choice of the parameter ν, a good trade-off can be achieved between the maximisation of fault detection probability and the minimisation of false alarm probability. In practice, the threshold values depend on the residual error amount due to the measurements errors, the model approximations and the disturbance signals that are not completely decoupled.

3.3 Fault Isolability and Structured Residual Set

The successful detection of an occurred fault is followed by the fault isolation procedure, which will distinguish (isolate) a particular fault from others. Whilst a single residual is sufficient to detect the occurrence of a fault, a set (or vector) of residuals is usually required for fault isolation. If a fault is distinguishable from other faults by exploiting a residual set, it can be said that this fault is isolable using this residual set. If the residual set can isolate all faults, it can be said that the residual set has the required isolability property (Chen and Patton 1999).

One approach to fulfil the fault isolation task consists in designing a set of structured residuals. Each residual in the set is designed to be sensitive to a subset of faults, whilst remaining insensitive to the remaining faults. The residual set, which has the required sensitivity to specific faults and insensitivity to other faults, is known as structured residual set (Gertler 1991; Chen and Patton 1999). The design procedure consists of two steps, the first step is to specify the sensitivity and insensitivity relationships between residuals and faults according to the assigned isolation task, and the second is to design a set of residual generators according to the desired sensitivity and insensitivity relationships. The exploitation of structured residual sets allows to simplify the diagnostic analysis to determining which of the residuals are non-zero.

The threshold test may be performed separately for each residual, yielding a Boolean decision table, and the isolation task can be fulfilled using this table and a proper decision logic.

In general, if all the possible faults $f_i(t),\ \ i=1,...,n_f$ are to be isolated, a residual set can be designed according to the following fault sensitivity conditions:

$$r_i(t) = R(f_i(t)),\quad i=1,...,n_f \tag{3.4}$$

where R denotes a functional relation. This is called as a *dedicated residual set*, and it is shown in Fig. 3.7.

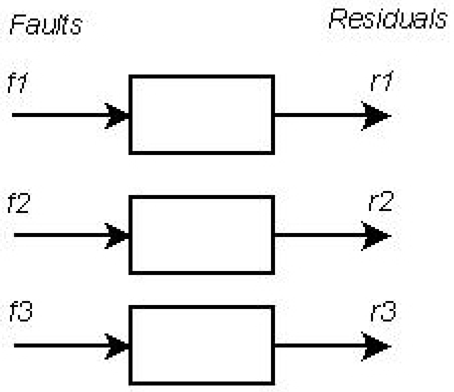


Figure 3.7 - Structured dedicated residual set for three distinct faults.

A simple threshold logic can be used to make decision about the occurrence of a specific fault by the logic decision according to

$$r_i(t) > T_i \Rightarrow f_i(t) \neq 0,\quad i=1,2,...,n_f \tag{3.5}$$

where $T_i,\ \ i=1,...,n_f$ are selected (fixed or adaptive) thresholds. This isolable residual structure is very simple and all faults can be detected simultaneously, however it is difficult to design in practice. Even when this structured residual set can be designed, there is normally no design freedom left to achieve other desirable performances such as robustness against modelling errors

(Wunnenberg 1990). A most commonly used and better scheme in designing the residual set is to make each residual sensitive to all but one fault, *i.e.*

$$\begin{cases} r_1(t) = R(f_2(t), \ldots, f_{n_f}(t)) \\ \vdots \\ r_i(t) = R(f_1(t), \ldots, f_{i-1}(t), f_{i+1}(t), \ldots, f_{n_f}(t)) \\ \vdots \\ r_{n_f}(t) = R(f_1(t), \ldots, f_{n_f-1}(t)) \end{cases} \tag{3.6}$$

This is defined as a *generalized residual set*, and it is shown in Fig. 3.8. If all residuals of the generalized residual set are generated using a bank of observers (observer-based residual generators), the structure is known as the generalized observer scheme (Frank 1987; Patton *et al.* 1989).

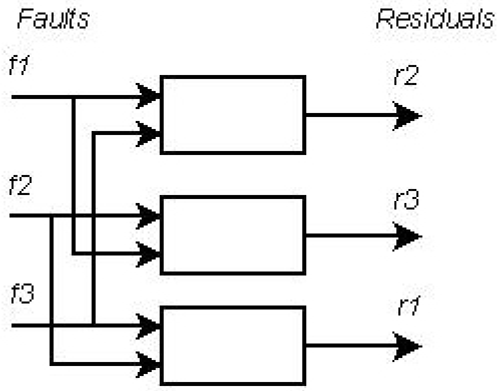


Figure 3.8 - Structured generalized residual set for three distinct faults.

The isolation task can again be performed using simple threshold testing according to the following logic scheme

$$\left.\begin{array}{l} r_i(t) \le T_i \\ r_j(t) > T_j \forall j \in \left\{1, \ldots, i-1, i+1, \ldots, n_f\right\} \end{array}\right\} \Rightarrow f_i(t) \ne 0, \quad i = 1, 2, \ldots, n_f \tag{3.7}$$

3.4 Fault Classification

In general, faults are events that can take place in different parts of the controlled system. In the FTC literature faults are usually classified according to different characteristics, *i.e.* faults location, fault type and time dependency. They can be classified according to their location of occurrence in the system, as also shown in Fig. 3.9, as follows (Edwards *et al.* 2010):

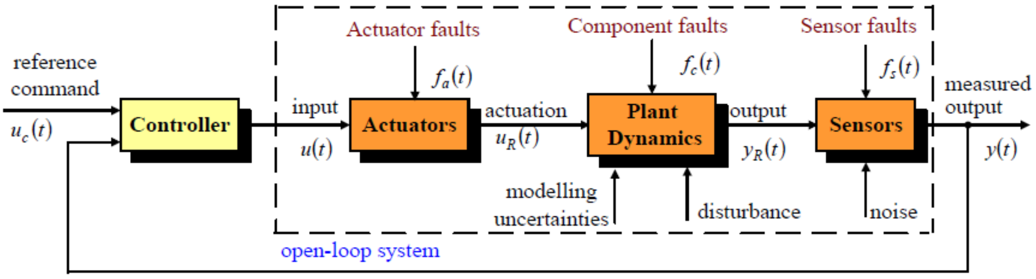


Figure 3.9 - Actuator fault, sensor fault and component faults (Sun 2013).

Actuator faults: they represent partial or total (complete) loss of control action effectiveness. An example of a completely lost actuator is a stuck actuator that produces no (controllable) actuation regardless of the input applied to it. Total actuator faults (*i.e.* failures) can occur, for instance, as a result of a breakage (the sticking of mechanical components or damage in the drive system *e.g.* due to bearings, gear wear or friction, caused by changes from the design characteristics or complete failure), cut or burned wiring, short circuits, or the presence of a foreign body in the actuator. Partially failed actuators produce only a part of the normal (*i.e.* under nominal operating conditions) actuation. This can result, for example, from jamming or oscillation, increased resistance or a fall in the supply voltage. Duplicating the actuators in the system in order to achieve increased fault-tolerance is often not an option due to their high prices and large size and mass.

This type of faults includes the following categories:

1. **Lock-in-place:** the actuator is stuck in a certain position/operating condition at an unknown time t_f and does not respond to subsequent commands:

$$u(t) = u(t_f) = \text{const}, \quad \forall t > t_f$$

2. **Failure:** the actuator produces zero force and torque, *i.e.* it becomes ineffective:

$$u(t) = 0, \quad \forall t > t_f$$

3. **Loss of effectiveness:** a decrease in the actuator gain that results in an actuated control input that is smaller than the commanded one:

$$u(t) = k \cdot u_c(t), \quad 0 < k < 1, \quad \forall t > t_f$$

where $u_c(t)$ stands for the required actuation, *i.e.* the commanded control input.

Sensor faults: these faults represent incorrect readings from the sensors that the system is equipped with. Sensor faults can also be subdivided into partial and total. Total sensor faults (*i.e.* failures) produce information that is not related to the value of the measured physical parameter. They can be due to broken communication wires (physical breakdown in a control loop) and loss of power. Partial sensor faults produce readings that are related to the measured signal in such a way that useful information could still be retrieved. This can, for instance, be an inaccurate calibration, a gain reduction (scaling error) so that a scaled version of the signal is measured, a biased measurement resulting in a (usually constant) offset in the reading, or increased noise (disturbance) in sensors. Due to their smaller sizes sensors can be duplicated in the system to increase fault tolerance. For instance, by using three sensors to measure the same variable one may consider it reliable enough to compare the readings from the sensors to detect faults in (one and only one) of them. The so-called majority voting method can then be used to pinpoint the faulty sensor. This approach usually implies significant increases in the related costs.

This type of faults includes the following categories:

1. **Lock-in-place:** the sensor output is stuck at a certain value at an unknown time t_f and does not provide the current value of the measured variable:

$$y(t) = y(t_f) = \text{const}, \quad \forall t > t_f$$

2. **Failure:** the sensor produces zero output, *i.e.* it becomes ineffective:

$$y(t) = 0, \quad \forall t > t_f$$

3. **Loss of effectiveness:** a degradation of the measurement accuracy of the sensor:

$$y(t) = k \cdot y_m(t), \quad 0 < k < 1, \quad \forall t > t_f$$

where $y_{true}(t)$ stands for the actual value of the measured variable.

Component faults: these are faults in the components of the plant itself, *i.e.* all faults that cannot be categorized as sensor or actuator faults will be referred to as component faults. These faults represent dynamical variations in the physical parameters of the system, *e.g.* mass, aerodynamic coefficients, damping constant, etc., that are often due to structural damage. They often result in a change in the dynamical behaviour of the controlled system. Due to their diversity, component faults cover a very wide class of (unanticipated) situations, and as such are the most difficult ones to deal with. In this thesis, component faults are not considered, and the proposed fault diagnosis and fault tolerant control schemes deals only with actuator and sensor faults in a spacecraft attitude control system.

Additionally, faults can be also classified according to their time dependency as abrupt, incipient and intermittent, as shown in Fig. 3.10 (Isermann and Ballé 1997; Edwards *et al.* 2010).

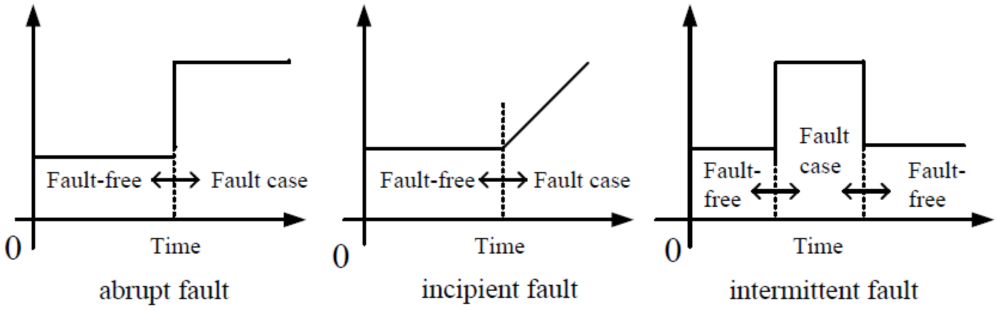


Figure 3.10 - Abrupt fault, incipient fault and intermittent fault (Sun 2013).

Abrupt faults: they occur instantaneously often as a result of hardware damage. These faults typically can be modelled as stepwise signals and can be represented by a sudden change in some actuator or sensor characteristic. Abrupt faults behave as variations that are faster than the nominal system dynamics, having a significant impact on the controlled system performance and/or stability. The effects of abrupt faults can be severe and hence the faults need to be detected and isolated quickly before they have an effect on system function and stability. However, they are relatively easy to diagnose.

Incipient faults: they represent slow parametric changes, often as a result of aging. They can be modelled by using ramp signals typically and represents an unexpected drift of the monitored signal. In contrast to an abrupt fault, an incipient fault (also known as a soft fault) has a very small but possibly slowly developing effect on the system and is very hard to detect and hence also to isolate. The incipient fault may not lead rapidly to serious consequence and be tolerable in their early stage, but may develop further in a continuous way into a more significant fault situation and hence must be detected and isolated (even identified) as quickly as possible to prevent system breakdown or any serious consequence if no prompt actions are taken.

Intermittent faults: these are not permanent faults that appear and disappear repeatedly, for instance due to partially damaged wiring (a loose connection in an electronic system).

Besides the aforementioned faults, there is another type of fault which is represented by a sinusoidal signal, known as an oscillatory fault. It can be considered as a type of incipient fault in terms of its time characteristics which behaves smoothly, not abruptly.

Finally, with respect to the way faults are modelled, they can be classified as additive type and multiplicative type (Edwards *et al.* 2010), as shown in Fig. 3.11. In general, additive faults are particularly suitable for representing component faults in the system, while sensor and actuator faults are in practice most often multiplicative by nature.

Additive faults: they behaves as an additional signal acting on the plant, for example, unexpected exogenous motion on an actuator.

Multiplicative faults: these faults are represented by the product of a system variable with the fault itself. They can appear as parametric deviation within a process, *i.e.* it may cause changes in the system dynamics.

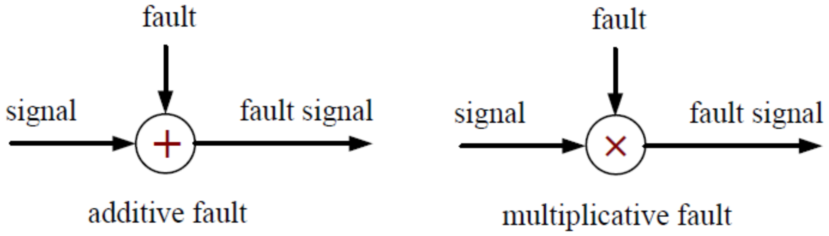


Figure 3.11 - Additive fault and multiplicative fault (Sun 2013).

It is worth noting that in this thesis, only additive fault representations are considered, due to the requirement of nonlinear system models affine both in the inputs and faults for the exploitation of the Nonlinear Geometric Approach (NLGA). Hence, multiplicative faults will be modelled in an equivalent but additive form.
Concerning the mathematical representation of these multiplicative and additive faults, multiplicative modelling usually is mostly used to represent sensor and actuator faults. Actuator faults represent malfunctioning of the actuators of the system. For simplicity of notation, the explicit indication of time dependency of the fault signals is omitted in the next mathematical relations. Such faults can be modelled as an abrupt or incipient change of the nominal (commanded) control action from u to the actual faulty (actuated) control action

$$u_f = u + (I - K_f^u)(\bar{u} - u) \tag{3.8}$$

where $\bar{u} \in \mathbb{R}^m$ is a (not necessarily constant) vector that cannot be manipulated, and where

$$K_f^u = diag\left\{\left[k_1^u, k_2^u, ..., k_m^u\right]\right\}, \quad k_i^u \in \mathbb{R}$$

In this way, for example, $k_i^u = 0$ represents a total fault (*i.e.* a complete failure) of the *i-th* actuator of the system so that the control action coming from this *i-th* actuator becomes equal to the *i-th* element of the uncontrollable offset vector $\bar{u}$, *i.e.* $u_{f,i} = \bar{u}_i$. On the other hand, $k_i^u = 1$ implies that the *i-th* actuator operates normally, *i.e.* $u_{f,i} = u_i$. The quantities k_i^u can also take values in between 0 and 1, making it possible to represent partial actuator faults.

Similarly, sensor faults occurring in the system represent incorrect reading from the sensors, so that as a result the actual output of the system differs from the variable being measured. The effect of multiplicative sensor faults can be modelled in the following way

$$y_f = y + (I - K_f^y)(\bar{y} - y) \tag{3.9}$$

where $\bar{y} \in \mathbb{R}^p$ is an offset vector, and

$$K_f^y = diag\left\{\left[k_1^y, k_2^y, ..., k_p^y\right]\right\}, \quad k_i^y \in \mathbb{R}$$

so that $k_j^y = 0$ represents a total fault of the *k-th* sensor, and $k_k^y = 1$ models the normal mode of operation of the *k-th* sensor. Partial faults are then modelled by taking k_k^y in between 0 and 1.

The multiplicative model is, usually, a natural way to model a wide variety of sensor and actuator faults, but cannot be used to represent more general component faults. Considering a generic input-affine nonlinear system affected by multiplicative sensor and actuator faults, it results that

$$\begin{cases} \dot{x} = g_0(x) + \sum_{i=1}^{m} g_i(x) u_{f,i} = g_0(x) + \sum_{i=1}^{m} g_i(x)(k_i^u u_i + (1 - k_i^u)\bar{u}_i) \\ y_f = K_f^y h(x) + (I - K_f^y)\bar{y} \end{cases} \tag{3.10}$$

On the other hand, the additive faults representation is more general than the multiplicative one. In the system model f_u, f_y are signals describing the faults on the system inputs and outputs (actuators and sensors). This representation may, in principle, be used to model a wide class of faults, including sensor, actuator, and component faults.

Considering a generic input-affine nonlinear system affected by additive sensor and actuator faults, it results that

$$\begin{cases} \dot{x} = g_0(x) + \sum_{i=1}^{m} g_i(x) u_i + \sum_{i=1}^{m} l_i(x) f_{u_i} \\ y = h(x) + \sum_{k=1}^{p} l_k(x) f_{y_k} \end{cases} \tag{3.11}$$

Actuator fault inputs and disturbances appear in the state dynamic equation and sensor faults in the output equation, respectively. However, using an additive fault representation in order to model faults such that they are equivalent to multiplicative faults of Eqs. (3.8) and (3.9), often results in signals f_u, f_y becoming related to one or more of the signals u, x and y. Just for example, when using this additive fault representation to model a total fault in an actuator ($k_i^u = 0$ and $u_{f,i} = \bar{u}_i = 0$

in Eq. (3.8)) then in order to make the model of Eq. (3.11) with additive faults equivalent to the model of Eq. (3.10), a signal $f_{u_i} = -u_i$ is needed, making f_u dependent on u. Therefore, both sensor and actuator additive faults can be modelled in a general way as functions of time, state (or output) and input as

$$f = f(t, x, u) \tag{3.12}$$

On the contrary, an advantage is, as already mentioned, that the additive representation can be used to model a more general class of faults than multiplicative ones. In addition, it is more suitable for the design of FDI/FDD schemes because the faults are represented by one signal rather than by changes in the dynamic model of the system, as is the case with the multiplicative representation. For that reason, the majority of FDI/FDD methods are focus on additive faults, as it will be shown for the NLGA (De Persis and Isidori 2001).

3.5 Modelling of Faults for the Design of the FDI System

A quite general state-space model for continuous-variable nonlinear system not affine in the states, inputs, faults and disturbances is given by

$$\begin{cases} \dot{x}(t) = g(x(t), u(t), d(t), f(t)) \\ y(t) = h(x(t), u(t), d(t), f(t)) \end{cases}$$

with initial condition $x(0) = x_0$. The NLGA and the FDI/FDD techniques used in the next sections and chapters of this dissertation (and several other model-based FDI/FDD methods) have been explicitly developed for nonlinear systems of the general form

$$\begin{cases} \dot{x} = g_0(x) + \sum_{i=1}^{m} g_i(x) u_i + \sum_{k=1}^{s} l_k(x) f_k + \sum_{j=1}^{r} p_j(x) d_j \\ y = h(x) \end{cases} \tag{3.13}$$

where the dynamics of the state $x \in \mathbb{R}^n$ (as well as the expression of the output $y \in \mathbb{R}^p$) are nonlinear in x, but affine in the control inputs $u_i,\ i = 1, \ldots, m$, in the additive fault inputs $f_k,\ k = 1, \ldots, s$, and in the disturbances $d_j,\ j = 1, \ldots, r$. Furthermore, control inputs, fault inputs and disturbances explicitly appear only in the state dynamics and not in the output equation. However, no assumption on the form and/or parameters of the fault time behaviour is required, in general.

As a consequence, FDI methods based on a model of the faulted process in the form of Eq. (3.13) are useful to deal with failures of hardware components (*e.g.*, actuators and/or sensors) of any type and time behaviour, but not affecting the structure of the system dynamics (*i.e.*, system faults are not easily treated).

For nonlinear systems in the form of Eq. (3.13), differential-geometric conditions have been given in De Persis and Isidori (2001), that are necessary for the solution of the FDI problem with possibly concurrent faults. These conditions, however, are violated in many situations of practical interest, notably whenever the total number of fault inputs exceeds the dimension of the state space. Several ways have been proposed to relax the FDI problem for system of Eq. (3.13) (Mattone and De Luca 2006a, 2006b), when it is not solvable in the original formulation of De Persis and Isidori (2001). One possibility is to introduce the additional assumption of no concurrency of faults, which results in much weaker necessary conditions for obtaining fault detection and isolation.

Considering a generic system possibly affected by faults affecting all the input signals $u_i,\ i=1,...,m$, and on the output measures $y_k,\ k=1,...,p$.

Hence, the total number of faults possibly affecting the system is given by $s=m+p$. When this number is larger than the dimension of the state space n, the necessary conditions for FDI given in De Persis and Isidori (2001) for possibly concurrent faults are certainly (and structurally) violated. Therefore, following the approach proposed by Mattone and De Luca (2006b), the FDI problem can be relaxed by assuming that at most one fault can affect the system at any time (non-concurring faults).

In the next sections, all steps involved in the nonlinear design of a general fault detection and isolation scheme for multiple non-concurring actuator and state sensor faults will be presented.

Hence, since the considered faults may affect all system inputs (actuators) and all available system outputs (state measurements), it is necessary to make some observations about the fault modelling for the design of the residual filters.

3.5.1 Modelling of Actuator Faults

Since the model of Eq. (3.13) is affine in the control inputs, actuator faults can be modelled in a straightforward manner through the fault input f_u defined as

$$f_u = u - u_c \tag{3.14}$$

where u_c is the commanded control input and u is the effectively actuated control input. Replacing the expression of Eq. (3.14) in the system state dynamics of Eq. (3.13) and neglecting for the moment the presence of disturbances, the system dynamics can be written as

$$\dot{x} = g_0(x) + \sum_{i=1}^{m} g_i(x) u_{c,i} + \sum_{i=1}^{m} l_{u_i}(x) f_{u_i} \tag{3.15}$$

with $l_{u_i}(x) = g_i(x)$. Eq. (3.15) includes the effects of all the actuator faults and is still affine in the (control and fault) inputs, thus, allowing the direct application of the nonlinear FDI methods illustrated in De Persis and Isidori (2001) and Mattone and De Luca (2006b).

3.5.2 Modelling of Sensor Faults

Focusing on the modelling of faults on the sensors providing measurements of the state variables $x_k,\ k=1,...,n$, the most natural way to take into account the possible occurrence of a fault would be defining a measurement fault F_{x_k} affecting the measurement of a generic *k-th* state variable x_k as the difference between the measured and real values of a state variable (*i.e.* $F_{x_k} = y_k - x_k$). However, this way of modelling would lead either to the appearance of the fault f_{x_k} in the output equation $y_k = x_k + F_{x_k}$ or to a model that is generally nonlinear in the sensor fault inputs (if the states in the model equations are replaced by $x_k = y_k - F_{x_k}$). In this condition it would not be followed the structure of (3.13) for the implementation of the NLGA in the design of the FDI residual filters described in the following subsections.

However, by using the procedure described in Mattone and De Luca (2006b) to model the sensor fault class in the considered spacecraft model, it would results in model equations with the input affine structure of Eq. (3.13).

In particular, a generic fault F_{x_k} on the output sensor of the *k-th* state variable can be described in an alternate way by a set of $\nu_k \geq 1$ mathematical fault inputs $f_{x_k,i}$ $(i = 1,...,\nu_k)$ in place of the natural fault F_{x_k}. However, in this way the one-to-one correspondence between the physical event (fault of a sensing device) and its mathematical representation (sensor fault input) is lost. Whenever a physical sensor fault occurs (*i.e.* $F_{x_k} \neq 0$), all the associated mathematical fault inputs $f_{x_k,i}$ will become generally nonzero, although with different time behaviours and, in general, without a direct physical interpretation. The detection of an occurred single physical fault F_{x_k} can be carried out by recognizing the occurrence of any (one or more) of the corresponding mathematical fault inputs $f_{x_k,i}$ $(i = 1,...,\nu_k)$.

Considering the generic *k-th* sensor fault ($k = 1,..., p$), the modelling procedure proposed in Mattone and De Luca (2006b) consists in the following steps:

1. Look in the system model for all different (and, in general, nonlinear) expressions $\varphi_{k,i}(x,u)$ involving x_k and such that the model is affine in $\varphi_{k,i}(x,u)$;
2. For each expression $\varphi_{k,i}(x,u)$, define the fault input $f_{x_k,i} = \varphi_{k,i}(x,u)\big|_{x_k = y_k} - \varphi_{k,i}(x,u)$, *i.e.* the error induced in the computation of $\varphi_{k,i}(x,u)$ by the use of the measured value y_k in place of the real value x_k, and compute the corresponding fault vector field $l_{x_k,i}(x)$. In this way, $\nu_k - 1$ faults are introduced. It is worth noting that, by definition, $f_{x_k,i}$ is only affected by a fault on the *k-th* state sensor (with the assumption of no concurrency of faults), and is zero whenever $F_{x_k} \neq 0$ (*i.e.* $x_k = y_k$). As a result of this modelling step, any occurrence of the expression $\varphi_{k,i}(x,u)$ in the system model can be replaced by $\varphi_{k,i}(x,u)\big|_{x_k = y_k} - f_{x_k,i}$, and the model results to be certainly affine in the fault input $f_{x_k,i}$. In this way, the right-hand side of Eq. (3.21) is only dependent on the variable y_k and not x_k;
3. Define the further fault input $f_{x_k,\nu_k} = \dot{y}_k - \dot{x}_k$. The introduction of this additional fault input in the model allows writing also the left-hand side of the *k-th* system equation in terms of the new variable y_k, with dynamics $\dot{y}_k = \dot{x}_k + f_{x_k,\nu_k}$. Thus, the fault vector field associated to f_{x_k,ν_k} is $l_{x_k,\nu_k} = I_k$, with I_k the *k-th* column of a $n \times n$ identity matrix;
4. If, for two indices *i*, *j*, it can be written $l_{x_k,i} = \alpha(x) l_{x_k,j}$ for some real function $\alpha(x)$, then it can be set $f_{x_k,j} = f_{x_k,j} + \alpha(x) f_{x_k,i}$ and eliminated $f_{x_k,i}$, whereas the vector field $l_{x_k,j}(x)$ clearly remains the same. With a slight abuse of notation, the symbol ν_k can be still used to indicate the final number of mathematical fault inputs corresponding to the *k-th* state sensor fault.

Therefore, when the outputs are taken as new state variables for the system dynamics, the general structure of Eq. (3.13) is recovered. The final model, including the effect of all (non-concurring) faults of actuators and state sensors, is then

$$\dot{y} = g_0(y) + \sum_{i=1}^{m} g_i(y) u_{c,i} + \sum_{i=1}^{m} l_{u_i}(y) f_{u_i} + \sum_{k=1}^{p} \sum_{j=1}^{\nu_k} l_{x_k,j}(y) f_{x_k,j} \qquad (3.16)$$

with the trivial output equation omitted. Eq. (3.16) models the faulted system, which is expressed in terms of the available commanded inputs u_c and measured outputs y, and is affine in all control and (unknown) fault inputs, as requested. In other words, any discrepancy between faultless and faulted system dynamics is fully summarized within the introduced fault inputs. Model of Eq. (3.16) may be reordered and more compactly rewritten as

$$\dot{y} = g_0(y) + \sum_{i=1}^{m} g_i(y) u_{c,i} + \sum_{k=1}^{s} \sum_{j=1}^{\mu_k} l_{k,j}(y) f_{k,j} \tag{3.17}$$

where m is the number of actuators (and actuator faults), p is the total number of sensor measurements (and sensor faults), $s = m + p$ is the total number of (physical) actuator and sensor faults, with $f_{k,1} = f_{u_k}$ and $\mu_k = 1$ for $k = 1,...,m$, and $f_{m+i,j} = f_{x_i,j}$ and $\mu_{m+i} = \nu_i$ for $i = 1,..., p$.

Finally, it is worth noting that input and measurement noise are not explicitly included in model of Eq. (3.17) used for FDI design. Indeed, it is rather intuitive (and can be formally proven) that these disturbances cannot be exactly distinguished from faults of the input actuators and state sensors. On the other hand, when these unmodelled disturbances are small enough, relevant faults can still be correctly isolated, in practice, by using suitable residual thresholds.

3.6 Nonlinear Geometric Approach

This section describes the general design methodology on which the residual generators for fault detection and isolation for nonlinear systems are based. This differential-geometric approach for nonlinear FDI problem, referred to as Nonlinear Geometric Approach (NLGA), was formally suggested by De Persis and Isidori (2000, 2001). It consists in finding changes of coordinates in the state space and in the output space, providing an observable quotient subsystem which, if it exists, is affected by a fault, but unaffected by disturbances and the other faults to be decoupled. In the following, necessary and sufficient conditions for the FDI problem to be solvable are given. A residual generator can be designed on the basis of the model of this observable subsystem.

As previously stated, the problem of fault detection and isolation in dynamical systems is the problem of generating diagnostic signals sensitive to the occurrence of faults. Regarding a fault as an input acting on the system, a diagnostic signal must be able to detect its occurrence, as well as to isolate this particular input from all other inputs (disturbances, controls, other faults) affecting the system behaviour.

One specific diagnostic signal (also called residual) must be generated per each fault to be detected, each diagnostic signal being sensitive only to one particular fault. Set in these terms, the problem of fault detection and isolation has very much the connotation of a problem of designing a system which, processing all available information about the plant, yields a non-interactive map between faults (viewed as inputs) and residuals (viewed as outputs).

In general, in Massoumnia (1986) and Massoumnia *et al.* (1989) it was proven that a basic necessary and sufficient condition for the problem to be solvable is the existence of an unobservability subspace (a subspace that can be made unobservable via output-injection and output-reduction) leading to a quotient (observable) subsystem unaffected by all fault signals but one. The subspace in question can be determined in a straightforward manner, from the parameters that characterize the plant, by means of simple recursive algorithms which were introduced earlier by Willems and Commault (1981) to a similar purpose (disturbance decoupling, with internal stability, via measurement feedback). Once this subspace has been determined, if the test of the necessary and sufficient condition is passed, the simple construction of an asymptotic observer for the quotient subsystem above yields the desired filter.

For the case of nonlinear systems, as shown in Isidori *et al.* (1981), to solve the problem of non-interacting control, the fault detection and isolation problem was approached in differential geometric terms. Hence, the solution to this problem can be characterized in terms of properties of certain distributions, which can be considered as the nonlinear analogue of the unobservability subspaces. As in the case of linear systems, the problem is solvable only if one of such distributions exists leading to a quotient system which is unaffected by all fault signals but one. Unobservability distributions can be computed by means of proper algorithms that extend to nonlinear systems those presented by Morse and Wonham (1970, 1971) and Willems and Commault (1981). Conversely, if such a distribution exists, it is possible to perform changes of coordinates (diffeomorphisms) in the state and in the output spaces which highlight a special internal structure and in particular the existence of a locally weakly observable subsystem, which is not affected by all fault signals but one. Once these changes of coordinates have been performed, the problem of designing a fault detection filter can be reduced to the design of an observer for the quotient subsystem.

A nonlinear system model affine with respect to the control and fault inputs and the disturbances is considered in the following form:

$$\begin{cases} \dot{x} = n(x) + g(x)u + \ell(x)f + p(x)d \\ y = h(x) \end{cases} \tag{3.18}$$

with initial conditions $x(0) = x_0$, in which the state vector $x \in \mathcal{X}$ (an open subset of $\mathbb{R}^n$), $u(t) \in \mathbb{R}^p$ is the control input vector, $f(t) \in \mathbb{R}$ is the fault, $d(t) \in \mathbb{R}^d$ the disturbance vector (embedding also the faults to be decoupled), and $y \in \mathbb{R}^m$ the output vector, $n(x)$, $l(x)$, the columns of $g(x)$ and $p(x)$ are smooth vector fields, $h(x)$ is a smooth map and $f(0) = 0$, $h(0) = 0$.

The three sets of components u, f, d of the input of Eq. (3.18) correspond, respectively, to an input channel to be used for control purposes, to a fault signal whose occurrence has to be detected, and to a disturbance signal, whose components include actual disturbances as well as other fault signals from which the specific fault has to be isolated. Hence, the main problem is the design of a filter (hereinafter referred to as the residual generator), modelled by equations of the form

$$\begin{cases} \dot{\hat{x}} = \hat{n}(\hat{x}, y) + \hat{g}(\hat{x}, y)u \\ r = \hat{h}(\hat{x}, y) \end{cases} \tag{3.19}$$

with state $\hat{x} \in \mathbb{R}^\nu$ defined in a neighbourhood $\hat{X}$ of the origin, inputs u, y and output $r \in \mathbb{R}^{\tilde{p}}$ with $p \le \tilde{p}$, in which $\hat{f}(\hat{x}, y)$, the m columns of $\hat{g}(\hat{x}, y)$ are smooth vector fields, $\hat{h}(\hat{x}, y)$ is a smooth mapping and $\hat{f}(0,0) = 0$, $\hat{h}(0,0) = 0$, such that the response $r(\cdot)$ of the cascaded system

$$\begin{cases} \dfrac{d}{dt}\begin{pmatrix} x \\ \hat{x} \end{pmatrix} = \begin{pmatrix} n(x) \\ \hat{n}(\hat{x}, h(x)) \end{pmatrix} + \begin{pmatrix} g(x) \\ \hat{g}(\hat{x}, h(x)) \end{pmatrix} u + \begin{pmatrix} l(x) \\ 0 \end{pmatrix} f + \begin{pmatrix} p(x) \\ 0 \end{pmatrix} d \\ r = \hat{h}(\hat{x}, h(x)) \end{cases} \tag{3.20}$$

depends nontrivially on (*i.e.* is affected by) the input f, depends trivially on (*i.e.* is decoupled from) the inputs u and d and asymptotically converges to zero whenever f is identically zero. In general, considering a nonlinear system affine in the inputs of the form

$$\begin{cases} \dot{x} = g_0(x) + \sum_{i=1}^{m} g_i(x)u_i \\ y = h(x) \end{cases} \tag{3.21}$$

with state $x \in \mathbb{R}^\nu$ defined in a neighbourhood X of the origin, $g_0(x)$, $g_1(x)$,..., $g_m(x)$ are smooth vector fields, $h(x)$ is a smooth function and Ω_O the smallest codistribution invariant under

$g_0,\ g_1,\ldots,\ g_m$ which contains $span\{dh\}$, the output of the system results to be decoupled from the generic input u_i if $g_i \in \Omega_{\mathrm{O}}^{\perp}$, and affected by u_i if $g_i \notin \Omega_{\mathrm{O}}^{\perp}$. Hence, considering the extended system

$$\begin{cases} \dot{x}^e = g_0^e(x^e) + \sum_{i=1}^{m} g_i^e(x^e)u_i + l^e(x^e)f + \sum_{i=1}^{d} p_i^e(x^e)d_i \\ r = h^e(x^e) \end{cases} \tag{3.22}$$

resulting from the composition of the plant and filter models, in which

$$\begin{aligned} & x^e = \begin{pmatrix} x \\ \hat{x} \end{pmatrix},\quad g_0^e(x^e) = \begin{pmatrix} f(x) \\ \hat{f}(\hat{x}, h(x)) \end{pmatrix},\quad g_i^e(x^e) = \begin{pmatrix} g(x) \\ \hat{g}(\hat{x}, h(x)) \end{pmatrix} \ \text{ for } i = 1,\ldots,m \\ & l^e(x^e) = \begin{pmatrix} l(x) \\ 0 \end{pmatrix},\quad p_i^e(x^e) = \begin{pmatrix} p(x) \\ 0 \end{pmatrix} \ \text{ for } i = 1,\ldots,d,\quad h^e(x^e) = \hat{h}(\hat{x}, h(x)) \end{aligned} \tag{3.23}$$

and looking for the smallest codistribution Ω_{O}^e containing $span\{dh\}$ and invariant under $g_i^e,\ i = 0,\ldots,m$ and all $p_i^e,\ i = 1,\ldots,d$, the output r of the system of Eq. (3.22) depends nontrivially on the input f if

$$l^e \notin \left(\Omega_{\mathrm{O}}^e\right)^{\perp} \tag{3.24}$$

and depends trivially on the inputs $u,\ d$ when $f = 0$ if

$$g_i^e \in \left(\Omega_{\mathrm{O}}^e\right)^{\perp},\quad \text{for } i = 1,\ldots,m \quad \text{and} \quad p_i^e \in \left(\Omega_{\mathrm{O}}^e\right)^{\perp},\quad \text{for } i = 1,\ldots,d \tag{3.25}$$

Therefore, the local Nonlinear Fundamental Problem of Residual Generation (lNLFPRG) consists in finding, if possible, a (stable) residual generator capable of detecting the occurrence of a specific exogenous (and unmeasured) input, such that the codistribution Ω_{O}^e in Eq. (3.22) satisfies

$$\begin{aligned} & span\{g_1^e,\ldots,\ g_m^e, p_1^e,\ldots,\ p_d^e\} \subset (\Omega_{\mathrm{O}}^e)^{\perp} \\ & span\{l^e\} \not\subset (\Omega_{\mathrm{O}}^e)^{\perp} \\ & \text{there is } \delta > 0 \text{ sucht hat, if } \left\|x^e(0)\right\| \le \delta, \text{ then } f(t) = 0 \text{ for all } t \ge 0 \Rightarrow \lim_{t\to\infty}\left\|r(t)\right\| = 0 \end{aligned} \tag{3.26}$$

In the same way, in general, diagnostic filters can be designed to detect and isolate multiple concurrent faults $f_1,\ldots,\ f_s$, in case of a system of the form

$$\begin{cases} \dot{x} = n(x) + \sum_{i=1}^{m} g_i(x)u_i + \sum_{i=1}^{s} l_i(x)f_i + p(x)d \\ y = h(x) \end{cases} \tag{3.27}$$

Setting $i = 1,\ldots,s$,

$$\tilde{p}_i = \begin{pmatrix} l_1 & \ldots & l_{i-1} & l_{i+1} & \ldots & l_s & p \end{pmatrix}$$

$$\tilde{d}_i = \begin{pmatrix} f_1 & \cdots & f_{i-1} & f_{i+1} & \cdots & f_s & d \end{pmatrix}$$

and writing (3.27) as

$$\begin{cases} \dot{x} = n(x) + \sum_{i=1}^{m} g_i(x) u_i + l_i(x) f_i + \tilde{p}(x)\tilde{d} \\ y = h(x) \end{cases} \tag{3.28}$$

Presuming that the residual generator

$$\begin{cases} \dot{\hat{x}}_i = \hat{n}_i(\hat{x}_i, y) + \hat{g}_i(\hat{x}_i, y) u \\ r_i = \hat{h}_i(\hat{x}_i, y) \end{cases} \tag{3.29}$$

solves the lNLFPRG problem for the generation of the residual for the fault f_i, then the bank of residual generators

$$\begin{cases} \dot{\hat{x}}_1 = \hat{n}_1(\hat{x}_1, y) + \hat{g}_1(\hat{x}_1, y) u \\ \quad = \ldots \\ \dot{\hat{x}}_s = \hat{n}_s(\hat{x}_s, y) + \hat{g}_s(\hat{x}_s, y) u \\ r_1 = \hat{h}_1(\hat{x}_1, y) \\ \quad = \ldots \\ r_s = \hat{h}_s(\hat{x}_s, y) \end{cases} \tag{3.30}$$

solves the problem of detecting and isolating each individual fault signal $f_1, \ldots, f_s$. Conversely, if there is a residual generator of the form

$$\begin{cases} \dot{\hat{x}} = \hat{n}(\hat{x}, y) + \hat{g}(\hat{x}, y) u \\ r_1 = \hat{h}_1(\hat{x}_1, y) \\ \quad = \ldots \\ r_s = \hat{h}_s(\hat{x}_s, y) \end{cases} \tag{3.31}$$

such that in the cascaded system of Eqs. (3.27) and (3.31), for each i, the residual r_i is affected by the input f_i and decoupled from the inputs u and $\tilde{d}_i$ and asymptotically convergent to zero as $f_i(\cdot)$ is identically zero, then necessarily the filter

$$\begin{cases} \dot{\hat{x}} = \hat{n}(\hat{x}, y) + \hat{g}(\hat{x}, y) u \\ r_i = \hat{h}_i(\hat{x}, y) \end{cases} \tag{3.32}$$

solves the lNLFPRG problem.

Now, in order to illustrate the NLGA algorithms exploited for the design of residual generators, let consider the system of the form of Eq. (3.21), where $g_0(x) = n(x)$ and $Ker\{dh\}$ is the distribution annihilating the differentials of the rows of the mapping $h(x)$, and a distribution Δ is said to be conditioned invariant ((h,n) invariant) if it satisfies

$$\left[g_i, \Delta \cap Ker\left\{dh\right\}\right] \subset \Delta \quad \text{for all } i = 0,...,m \tag{3.33}$$

Given an additional set of smooth vector fields $p_1(x),..., p_d(x)$ (corresponding to the inputs to be decoupled), and once defined the distribution (decoupling matrix)

$$P = span\left\{p_1,..., p_d\right\} \tag{3.34}$$

it can be considered the non-decreasing sequence of distributions defined as follows

$$\begin{gathered} S_0 = \overline{P} \\ S_{k+1} = \overline{S}_k + \sum_{i=0}^{m}\left[g_i, \overline{S}_k \cap Ker\left\{dh\right\}\right] \end{gathered} \tag{3.35}$$

where $\overline{S}$ denotes the involutive closure of S. If it exist an integer $k^* \geq 0$ such that $S_{k^*+1} = \overline{S}_{k^*}$ (well-defined distribution) the recursive algorithm stops and the minimal conditioned invariant distribution $\Sigma_*^P = \overline{S}_{k^*}$ is defined, which contains P and results to be involutive and conditioned invariant. Moreover, any other involutive distribution Δ containing P and conditioned invariant satisfies $\Delta \supset \Sigma_*^P$. A codistribution Ω is said to be conditioned invariant if it satisfies

$$L_{g_i}\Omega \subset \Omega + span\{dh\} \quad \text{for all } i = 0,...,m \tag{3.36}$$

where $span\{dh\}$ is the codistribution spanned by the differentials of the rows of the mapping $h(x)$. If $\Delta \cap Ker\left\{dh\right\}$ is a smooth distribution and $\Omega = \Delta^{\perp}$ is a smooth codistribution, then Ω satisfies (3.36). If Σ_*^P is well-defined and nonsingular and $\Sigma_*^P \cap Ker\left\{dh\right\}$ is a smooth distribution, it can be asserted that $\left(\Sigma_*^P\right)^{\perp}$ is the maximal conditioned invariant codistribution which is locally spanned by exact differentials and contained in $P^{\perp}$.

Considering again the system of Eq. (3.21) and given a fixed codistribution Θ, a non-decreasing sequence of codistributions can be defined as follows

$$\begin{gathered} Q_0 = \Theta \cap span\left\{dh\right\} \\ Q_{k+1} = \Theta \cap \left(\sum_{i=0}^{m} L_{gi}Q_k + span\left\{dh\right\}\right) \end{gathered} \tag{3.37}$$

Supposing that all codistributions of this sequence are nonsingular, so that there is an integer $k^* \leq n-1$ such that $Q_k = Q_{k^*}$ for all $k > k^*$, the recursive algorithm stops and the codistribution $\Omega^* = Q_{k^*}$ can be defined. It is convenient to use the notation $\Omega^* = o.c.a.(\Theta)$ (where o.c.a. stands for observability codistribution algorithm) to stress the dependency on Θ of the codistribution $\Omega^* = Q_{k^*}$.

Supposing that all codistributions of the generated sequence are nonsingular and let $\Omega^* = o.c.a.(\Theta)$ be defined as above, then

$$\begin{aligned} Q_0 &= \Omega^* \cap span\{dh\} \\ Q_{k+1} &= \Omega^* \cap \left(\sum_{i=0}^{m} L_{gi} Q_k + span\{dh\} \right) \end{aligned} \tag{3.38}$$

As a consequence $o.c.a.(\Omega^*) = \Omega^*$. Moreover, if the codistribution Θ is conditioned invariant, so it is the codistribution Ω^*. Hence, a generic codistribution Ω is a observability codistribution for the system (3.21) if

$$\begin{aligned} & L_{g_i}\Omega \subset \Omega + span\{dh\} \quad \text{for all } i = 0,...,m \\ & o.c.a.(\Omega) = \Omega \end{aligned} \tag{3.39}$$

Likewise, Δ is an unobservability distribution if its annihilator $\Omega = \Delta^\perp$ is an observability codistribution. If Θ is a conditioned invariant distribution, then $o.c.a.(\Theta)$ is an observability codistribution.

If the recursive algorithm of Eq. (3.37) is initialized at $\left(\Sigma_*^P\right)^\perp$, then $o.c.a.\left(\left(\Sigma_*^P\right)^\perp\right)$ is by construction an observability codistribution contained in $P^\perp$. As a matter of fact, $o.c.a.\left(\left(\Sigma_*^P\right)^\perp\right)$ is the largest codistribution having such property. The codistribution $o.c.a.(\Theta)$ is the maximal (in the sense of codistribution inclusion) observability codistribution contained in Θ. If Σ_*^P is well-defined and nonsingular, and $\Sigma_*^P \cap Ker\{dh\}$ is a smooth distribution, then $o.c.a.\left(\left(\Sigma_*^P\right)^\perp\right)$ is the maximal (in the sense of codistribution inclusion) observability codistribution which is locally spanned by exact differentials and contained in $P^\perp$.

Finally, considering again the system of Eq. (3.21) and the decoupling matrix $P = span\{p_1,..., p_d\}$, if $l_i(x) \notin o.c.a.\left(\left(\Sigma_*^P\right)^\perp\right)^\perp = o.c.a.\left(\Omega^*\right)^\perp$, the fault f_i results to be detectable and a change of coordinates can be determined. Whenever the previous fault detectability condition is satisfied, there exists a surjection Ψ_1 and a function Φ_1 which satisfy $\Omega^* \cap span\{dh\} = span\{d(\Psi_1 \circ h)\}$ and $\Omega^* = span\{d\Psi_1\}$ respectively. Therefore, the functions $\Psi(y)$ and $\Phi(x)$ can be defined as

$$\Psi(y(x)) = \begin{pmatrix} \bar{y}_1 \\ \bar{y}_2 \end{pmatrix} = \begin{pmatrix} \Psi_1(y) \\ H_2 y \end{pmatrix}, \quad \Phi(x) = \begin{pmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \end{pmatrix} = \begin{pmatrix} \Phi_1(x) \\ H_2 h(x) \\ \Phi_3(x) \end{pmatrix} \tag{3.40}$$

with

$$\begin{gathered} n_1 = \dim\{\Omega^*\}, \quad n_2 = p - n_1 \\ \dim\{\Omega^* \cap span\{dh\}\} = p - n_2 \\ \Psi_1(y) : \mathbb{R}^p \to \mathbb{R}^{p-n_2}, \quad \Phi_1(x) : \mathbb{R}^n \to \mathbb{R}^{n_1} \\ H_2 y : \mathbb{R}^p \to \mathbb{R}^{n_2}, \qquad \Phi_3(x) : \mathbb{R}^n \to \mathbb{R}^{n-n_1-n_2} \end{gathered} \tag{3.41}$$

where H_2 is a selection matrix (*i.e.* a matrix in which any row has all 0 entries but one, which is equal to 1). $\bar{x}_1$ represents the measured part of the state which is affected by the fault f_i and not affected by the other faults and disturbances $\tilde{d}_i$, $\bar{x}_2$ and $\bar{x}_3$ represent the measured and unmeasured parts of the state affected by the fault f_i and the other inputs $\tilde{d}_i$ respectively. In many cases $\bar{x}_3$ is not present. The functions $\Psi(y)$ and $\Phi(x)$ are (local) diffeomorphisms at $x^0 \in X$ and $y^0 = h(x^0) \in \mathbb{R}^p$.

Thus, in the new (local) coordinates defined above, the system of Eq. (3.18) can be described by the following relations (Benini *et al.* 2008):

$$\begin{cases} \dot{\bar{x}}_1 = n_1(\bar{x}_1,\bar{x}_2) + g_1(\bar{x}_1,\bar{x}_2)u + l_1(\bar{x}_1,\bar{x}_2,\bar{x}_3)f \\ \dot{\bar{x}}_2 = n_2(\bar{x}_1,\bar{x}_2,\bar{x}_3) + g_2(\bar{x}_1,\bar{x}_2,\bar{x}_3)u + \\ \quad +l_2(\bar{x}_1,\bar{x}_2,\bar{x}_3)f + p_2(\bar{x}_1,\bar{x}_2,\bar{x}_3)d \\ \dot{\bar{x}}_3 = n_3(\bar{x}_1,\bar{x}_2,\bar{x}_3) + g_3(\bar{x}_1,\bar{x}_2,\bar{x}_3)u + \\ \quad +l_3(\bar{x}_1,\bar{x}_2,\bar{x}_3)f + p_3(\bar{x}_1,\bar{x}_2,\bar{x}_3)d \\ \bar{y}_1 = h(\bar{x}_1) \\ \bar{y}_2 = \bar{x}_2 \end{cases} \tag{3.42}$$

with $l_1(\bar{x}_1,\bar{x}_2,\bar{x}_3)$ not identically zero. If $\bar{x}_2$ is identified with $\bar{y}_2$ and viewed as an independent input, the $\bar{x}_1$-subsystem can be singled out:

$$\begin{cases} \dot{\bar{x}}_1 = n_1(\bar{x}_1,\bar{y}_2) + g_1(\bar{x}_1,\bar{y}_2)u + l_1(\bar{x}_1,\bar{y}_2,\bar{x}_3)f \\ \bar{y}_1 = h(\bar{x}_1) \end{cases} \tag{3.43}$$

state which is affected by the fault f_i and not affected by the other faults and disturbances $\tilde{d}_i$ (embedding both the other faults and disturbances). Therefore, in general, the NLGA residual generators for FDI can be designed by exploiting the properties of this quotient subsystem.

3.7 Residual Generator Design

The additional assumption of no concurrency of faults, made in Section 3.5 in order to consider also sensor faults, results into much weaker conditions for fault detectability and isolability than those given in De Persis and Isidori (2001) and described above. In particular, when each physical fault F_k is modelled by just one fault input with associated vector field, it has been shown that the necessary and sufficient condition for non-concurring FDI (under full state availability and absence of disturbances) is

$$span\{l_i\} \not\subseteq span\{l_k\}, \quad \forall i,k, \quad i \neq k \tag{3.44}$$

Condition (3.44) guarantees that, for each couple of faults F_i, F_k, a dynamic system (residual generator) can be found, whose output r (residual) is certainly affected by just one of the two faults and not by the other, thus allowing the discrimination of the occurrence of one fault from another one. When each physical fault F_k is modelled by a set of always concurrent fault inputs, as in the case of Eq. (3.17), condition (3.44) can be easily extended, as described in Mattone and De Luca (2006a, 2006b), by observing that a residual is:

1. affected by F_k, if it is affected by at least one of the associated fault inputs $f_{k,1},..., f_{k,\mu_k}$;
2. decoupled from F_k, if it is not affected by any of the fault inputs $f_{k,1},..., f_{k,\mu_k}$.

This leads to the following necessary and sufficient condition for detectability and isolability

$$\begin{gathered}\forall k, \forall i \neq k, \exists j \in \{1,...,\mu_k\} \text{ such that } span\{l_{k,j}\} \underline{\not\subset} \bar{P}_i \\ OR \\ \exists h \in \{1,...,\mu_i\} \text{ such that } span\{l_{i,h}\} \underline{\not\subset} \bar{P}_k\end{gathered} \tag{3.45}$$

where $P_i = span\{l_{i,1},...,l_{i,\mu_i}\}$ and $\bar{P}_i$ denotes the involutive closure of P_i, *i.e.* the closure of P_i under the Lie bracket operator. It is worth noting that, differently from Eq. (3.44), which is symmetric with respect to i and k, the two conditions in the left-hand and right-hand sides of the *OR* operator in Eq. (3.45) may not hold at the same time. Thus, it may happen that a residual generator exists, that is affected by F_i and not by F_k, but that any residual affected by F_k is necessarily also affected by F_i. The fulfilment of the condition (3.45) of detectability and isolability implies that for any pair of faults F_i and F_k, a residual exists that is affected by F_i or by F_k, but not by both. Moreover, the fulfilment of this condition implies there exists a set R of residuals such that each fault affects a different, nonempty subset of diagnostic signals within R. Under the assumption of full state measurability, the design of this set of residuals follows directly from Eq. (3.45), using the fault vector fields of model of Eq. (3.17). Essentially, condition (3.45) guarantees the existence, for any pair (*i*,*k*), of a suitable decoupling output function $\Psi_1(y) = \bar{y}_1$, whose dynamics is affected, for example, by at least one of the fault inputs associated to F_i, and decoupled from all fault inputs corresponding to F_k. Then, a residual generator can be designed as a standard nonlinear observer with linear error dynamics (Isidori 1995). In particular, the dynamics of the state of this observer/residual generator is a copy of the nominal (faultless) dynamics of $\Psi_1(y) = \bar{y}_1$, plus a correction term, *i.e.* the residual feedback, that makes the observation error/residual asymptotically converge to zero in the absence of faults. If the condition

$$\exists h \in \{1,...,\mu_i\} \text{ such that } span\{l_{i,h}\} \underline{\not\subset} \bar{P}_k \tag{3.46}$$

holds, since $\bar{P}_k$ is involutive by construction, $n - \dim(\bar{P}_k)$ independent functions $\psi_{\mathrm{k},n-\dim(\bar{P}_k)}(y)$ certainly exist, such that their differentials $d\psi_{k,1}(y),...,\psi_{\mathrm{k},n-\dim(\bar{P}_k)}(y)$ span the annihilating codistribution $\bar{P}_k^{\perp}$ (Isidori 1995), with

$$\frac{\partial \psi_{k,j}}{\partial y} \upsilon = 0 \qquad \forall j = 1,...,n - \dim(\bar{P}_k),\ \forall \upsilon \in \bar{P}_k \tag{3.47}$$

Moreover, from Eq. (3.45) it follows that

$$\exists l \in \{1,...,n - \dim(\bar{P}_k)\} \text{ such that } \frac{\partial \psi_{k,l}}{\partial y} l_{i,h} \neq 0 \tag{3.48}$$

and a scalar output $\bar{y}_{1s} = \psi_{k,l}(y)$ can be determined, whose evolution is defined by

$$\dot{\bar{y}}_{1s} = \frac{\partial \psi_{k,l}}{\partial y}\left(g_0(y) + \sum_{j=1}^{m} g_j(y) u_{c,j} + \sum_{p=1}^{s}\sum_{q=1}^{\mu_p} l_{p,q}(y)\, f_{p,q} \right) \tag{3.49}$$

and, thus, is affected by the physical fault F_i, in particular the associated fault input $f_{i,\mathrm{h}}$, and not affected by F_k, since Eq. (3.47) holds for $\upsilon = f_{k,q}, q = 1,...,\mu_k$ in particular. The residual generator allowing to discriminate F_i from F_k is modelled as

$$\begin{cases} \dot{\xi} = \dfrac{\partial \psi_{k,l}}{\partial y}\left(g_0(y) + \displaystyle\sum_{j=1}^{m} g_j(y) u_{c,j} \right) + K(\bar{y}_{1s} - \xi) \\ r = \bar{y}_{1s} - \xi \end{cases} \tag{3.50}$$

with $K > 0$. This residual generator is structured as a nonlinear observer with scalar state ξ and linear error dynamics (Isidori 1995; Benini *et al.* 2008). The residual is characterized by the dynamics

$$\dot{r} = -Kr + K\sum_{p=1}^{s}\sum_{q=1}^{\mu_p} l_{p,q}(y)\, f_{p,q}$$

of a linear, exponentially stable system driven by the set of all the fault inputs $f_{p,q}$ for which it holds

$$\frac{\partial \psi_{k,l}}{\partial y} l_{p,q} \neq 0$$

By construction, the set of these fault inputs certainly includes $f_{i,\mathrm{h}}$, but does not include any fault input $f_{k,j}, j = 1,...,\mu_k$, even if it may or not be affected by all the other faults.

Now, in order to design the set R of residual filters for the FDI, it could be more convenient to work with residual matrices, due to the possible large number of mathematical fault inputs associated to the physical faults (Mattone and De Luca 2006b).

The Residual Matrix (RM) associated to the set $R = \{r_1,...,\mathrm{r}_{\mathrm{n_r}}\}$ of n_r residuals is a binary matrix whose element $\mathrm{RM}(i, j)$ is nonzero if and only if the physical fault F_i (or, equivalently, at least one of the associated fault inputs $f_{i,1},..., f_{i,\nu_i}$) affects residual $r_j \in R$. Therefore, a set of residuals solves the non-concurrent FDI problem if and only if

$$\mathrm{RM}(i,:) \neq 0 \quad \mathrm{RM}(k,:) \neq \mathrm{RM}(i,:) \quad \forall \mathrm{i,k,\ i} \neq \mathrm{k} \tag{3.51}$$

where $\mathrm{RM}(i,:)$ denotes the *i-th* row of RM. The incremental procedure described in Mattone and De Luca (2006b) to build the columns of the residual matrix RM, which satisfies condition (3.51), can be summarized as follows:

1. Initially, the first element r_1 of the residual set R is designed, so that it discriminates the physical fault F_1 from F_2, and the corresponding residual matrix RM, consisting of a single column, is built. After this initial step, the first two rows of RM are different, but one of them is zero.

2. Then the element r_2 of the residual set is designed, so that the first two rows of the resulting residual matrix RM, now consisting of two columns, become nonzero. From now on, the first two rows of RM will certainly be different and nonzero for whatever selection of the remaining elements of the residual set.
3. After the generic *k-th* step of the procedure, the first *k* rows of the current RM matrix are different and nonzero. Then, looking at rows from *k+1* to the total number $s = m + p$ of physical faults possibly affecting the system, the first row being either zero, or equal to one of the above matrix rows (and, necessarily, only to one), is founded (if it exists). Consequently, the residual r_{k+1} to be added to the set is designed so as to introduce either a one element in the null row, or the element needed to make the two equal rows to be different.
4. After a finite number of n_r steps, with $\log_2 s \leq n_r \leq s$, all the *s* rows of RM will certainly be different and nonzero, and the designed residuals will satisfy condition (3.51).

Moreover, the procedure described above can be also extended to the case in which external disturbances to be decoupled are present. In this case, the disturbance term d in the affine nonlinear model of Eq. (3.18) can be considered as a vector of additional and always concurrent fault inputs to which any designed residual filter has to be completely insensitive, *i.e.* all the designed residual filters have to be always not sensitive to all the inputs included in d . In this way, the total number of faults possibly affecting the system results to be $s = m + p + n_d$, where n_d is the number of elements in the disturbance vector d . Now, the main difference in the design procedure of the residual filters consists in the determination of a residual matrix in which the rows corresponding to the disturbance terms to be decoupled are zero.

If the condition (3.46) holds, where in this case $P_i = span\{l_{i,1},...,l_{i,\mu_i}, p\}$, p is the set of smooth vector fields corresponding to the concurring inputs to be always decoupled and $\overline{P}_i$ still denotes the involutive closure of P_i. Therefore, any residual filter designed by means of the previously described incremental procedure and satisfying, for example, condition (3.46) for an extended distribution $P_i = span\{l_{i,1},...,l_{i,\mu_i}, p\}$ results to be decoupled from the disturbance vector d . Model (3.16) may be rewritten to include also the (always concurrent) disturbance inputs as

$$\dot{y} = g_0(y) + \sum_{i=1}^{m} g_i(y)u_{c,i} + \sum_{k=1}^{s}\sum_{j=1}^{\mu_k} l_{k,j}(y)\, f_{k,j} + \sum_{q=1}^{n_d} p_q(y)d_q \tag{3.52}$$

where m is the number of actuators (and actuator faults), p is the total number of sensor measurements (and sensor faults), $s = m + p + n_d$ is the total number of (physical) actuator and sensor faults and disturbance inputs.

4 FDI Implementation in the Satellite ADCS

This chapter illustrates the design and practical implementation of a complete FDI system for the considered spacecraft ADCS. The FDI system consists of four independent banks of residual generators, referred to as Residual Banks (RB)s from now on, working in parallel and designed by exploiting the NLGA and the previously described fault modelling procedure and decoupled from the external aerodynamic disturbance torque acting on the satellite. Subsequently, the fault detection and isolation procedure, based on the cross-check of the provided diagnostic signals, is described along with the corresponding decision logic. Simulation results for FDI are given in case of sensor and actuator fault occurrence.

4.1 Mathematical Fault Inputs Associated to Physical Faults

In order to design a set of residual generators for fault detection and isolation purpose, the previously described procedure for fault mapping is implemented and the mathematical fault inputs associated to each physical actuator and sensor fault are determined. Considering these associated fault inputs, the original nonlinear model can be rewritten in an affine form with respect to both the inputs and fault inputs, and the NLGA can be subsequently exploited in order to design proper residual filters, each of them sensitive only to a specific set of possible mathematical faults and not sensitive to the external aerodynamic disturbance. As a consequence, these designed residual filters provide diagnostic signals that can be exploited to realize an accurate fault isolation.

For simplicity, from now on the nonlinear dynamic and kinematic model equations (2.46) and (2.33) of the spacecraft, and the dynamic equations (2.42) of the reaction wheels will be written with a different notation, by considering the following notations for actuated input vector $\mathbf{u}\in\mathbb{R}^4$, commanded input vector $\mathbf{u}_c\in\mathbb{R}^4$, state vector $\mathbf{x}\in\mathbb{R}^{11}$ and output vector $\mathbf{y}\in\mathbb{R}^{15}$:

$$\mathbf{u}=\begin{bmatrix}u_1 & u_2 & u_3 & u_4\end{bmatrix}^T=\begin{bmatrix}T_{ctrl,1} & T_{ctrl,2} & T_{ctrl,3} & T_{ctrl,4}\end{bmatrix}^T \tag{4.1}$$

$$\mathbf{u}_c=\begin{bmatrix}u_{c1} & u_{c2} & u_{c3} & u_{c4}\end{bmatrix}^T=\begin{bmatrix}T_{c,1} & T_{c,2} & T_{c,3} & T_{c,4}\end{bmatrix}^T \tag{4.2}$$

$$\begin{aligned}\mathbf{x}&=\begin{bmatrix}x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11}\end{bmatrix}^T=\\&=\begin{bmatrix}\omega_{w,1} & \omega_{w,2} & \omega_{w,3} & \omega_{w,4} & \omega_{x,i} & \omega_{y,i} & \omega_{z,i} & q_{1,o} & q_{2,o} & q_{3,o} & q_{4,o}\end{bmatrix}_{true}^T\end{aligned} \tag{4.3}$$

$$\begin{aligned}\mathbf{y}&=\begin{bmatrix}y_1 & y_2 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 & y_9 & y_{10} & y_{11} & y_{12} & y_{13} & y_{14} & y_{15}\end{bmatrix}^T=\\&=\begin{bmatrix}\omega_{w,1} & \omega_{w,2} & \omega_{w,3} & \omega_{w,4} & \omega_{x,i} & \omega_{y,i} & \omega_{z,i} & q_{star1,1} & q_{star1,2} & q_{star1,3} & q_{star1,4} & q_{star2,1} & q_{star2,2} & q_{star2,3} & q_{star2,4}\end{bmatrix}_{measured}^T\end{aligned} \tag{4.4}$$

where $\bar{\mathbf{q}}_{star1}=[q_{star1,1}\ q_{star1,2}\ q_{star1,3}\ q_{star1,4}]_{measured}^T$ and $\bar{\mathbf{q}}_{star2}=[q_{star2,1}\ q_{star2,2}\ q_{star2,3}\ q_{star2,4}]_{measured}^T$ are the attitude measurements provided by the two considered attitude sensors, *i.e.* the two star trackers. The input vectors consist of the actuated (true) and commanded (nominal) control torques provided by the four reaction wheels in the tetrahedron configuration, respectively. The state vector consists of the value of the (true) spin rates of the actuator flywheels, the angular velocities of the satellite in the inertial reference frame along its body axes and the true attitude unit quaternion $\bar{\mathbf{q}}_o=\begin{bmatrix}q_{1,o} & q_{2,o} & q_{3,o} & q_{4,o}\end{bmatrix}^T$ in the orbital reference frame. All the states are assumed measurable, or derivable by means of Eq. (2.57) from the sensor measurements, thus the output vector consists of the measured values of all the states provided by the considered sensors.

With this notation, the differential equations (2.34), (2.42) and (2.46) of the spacecraft nonlinear model can be rewritten as follows. The dynamic equations (2.42) of the actuators are explicitly rewritten as

$$\begin{cases} \dot{x}_1 = -\dfrac{u_1}{I_w} \\ \dot{x}_2 = -\dfrac{u_2}{I_w} \\ \dot{x}_3 = -\dfrac{u_3}{I_w} \\ \dot{x}_4 = -\dfrac{u_4}{I_w} \end{cases} \tag{4.5}$$

The dynamic equations (2.46) of the satellite attitude are explicitly rewritten as

$$\begin{cases} \dot{x}_5 = -\dfrac{(I_{zz}-I_{yy})}{I_{xx}}x_6x_7 + \sqrt{\dfrac{2}{3}}\dfrac{I_w}{I_{xx}}(x_7x_1 - x_7x_2 - x_6x_4 + x_6x_3) + \\ \quad +\sqrt{\dfrac{1}{3}}\dfrac{(u_1+u_2-u_3-u_4)}{I_{xx}} + \dfrac{6\mu}{R^3}\dfrac{(I_{zz}-I_{yy})}{I_{xx}}(x_9x_{10}+x_8x_{11})(1-2x_8^2-2x_9^2) + \\ \quad +\dfrac{F_{aero}}{I_{xx}}(2x_8x_{10}c_{p,y} + 2x_9x_{11}c_{p,y} - 2x_8x_9c_{p,z} + 2x_{10}x_{11}c_{p,z}) \\ \dot{x}_6 = -\dfrac{(I_{xx}-I_{zz})}{I_{yy}}x_5x_7 + \sqrt{\dfrac{1}{3}}\dfrac{I_w}{I_{yy}}(\sqrt{2}x_5x_4 - \sqrt{2}x_5x_3 - x_7x_1 - x_7x_2 + x_7x_3 + x_7x_4) + \\ \quad +\sqrt{\dfrac{2}{3}}\dfrac{(u_1-u_2)}{I_{yy}} + \dfrac{6\mu}{R^3}\dfrac{(I_{xx}-I_{zz})}{I_{yy}}(x_8x_{10}-x_9x_{11})(1-2x_8^2-2x_9^2) + \\ \quad +\dfrac{F_{aero}}{I_{yy}}(c_{p,z} - 2x_9^2c_{p,z} - 2x_{10}^2c_{p,z} - 2x_8x_{10}c_{p,x} - 2x_9x_{11}c_{p,x}) \\ \dot{x}_7 = -\dfrac{(I_{yy}-I_{xx})}{I_{zz}}x_5x_6 + \sqrt{\dfrac{1}{3}}\dfrac{I_w}{I_{zz}}(x_6x_1 + x_6x_2 - x_6x_3 - x_6x_4 - \sqrt{2}x_5x_1 + \sqrt{2}x_5x_2) + \\ \quad +\sqrt{\dfrac{2}{3}}\dfrac{(u_4-u_3)}{I_{zz}} + \dfrac{12\mu}{R^3}\dfrac{(I_{yy}-I_{xx})}{I_{zz}}(x_9x_{10}+x_8x_{11})(x_8x_{10}-x_9x_{11}) + \\ \quad +\dfrac{F_{aero}}{I_{zz}}(2x_8x_9c_{p,x} - 2x_{10}x_{11}c_{p,x} - c_{p,y} + 2x_9^2c_{p,y} + 2x_{10}^2c_{p,y}) \end{cases} \tag{4.6}$$

Finally, the kinematic equations (2.34) of the satellite attitude are explicitly written as:

$$\begin{cases} \dot{x}_8 = \frac{1}{2}(x_5 x_{11} - x_6 x_{10} + x_7 x_9 + \omega_o x_{10}) \\ \dot{x}_9 = \frac{1}{2}(x_5 x_{10} + x_6 x_{11} - x_7 x_8 + \omega_o x_{11}) \\ \dot{x}_{10} = \frac{1}{2}(-x_5 x_9 + x_6 x_8 + x_7 x_{11} - \omega_o x_8) \\ \dot{x}_{11} = \frac{1}{2}(-x_5 x_8 - x_6 x_9 - x_7 x_{10} - \omega_o x_9) \end{cases} \tag{4.7}$$

The output equations, considering also the presence of a second redundant attitude sensor, are defined as

$$\mathbf{y} = h(\mathbf{x}) = \begin{bmatrix} I_7 & 0 \\ 0 & I_4 \\ 0 & I_4 \end{bmatrix} \mathbf{x} \tag{4.8}$$

Now, it is worth recalling that the attitude and angular velocity sensors considered in this thesis actually provide measurements of the satellite attitude and angular rates of the satellite with respect to the inertial reference frame. However, it is worth noting that, in order to maintain a unique notation for the state variables in the residual filter models, and simplify the definition of the associated mathematical fault inputs, the Eqs. (4.6) and (4.7) in this chapter have been rewritten considering the spacecraft quaternion vector defined with respect to the orbital frame as state variable of the dynamic model. The values of the corresponding output variables can be computed at each simulation instant by exploiting the actual sensor outputs for the spacecraft attitude provided by the two star trackers with respect to the inertial reference frame of the fixed stars in Eq. (2.57), with the position of the spacecraft along the orbit assumed to be exactly known. On the other hand, it is worth noting that the spacecraft angular velocity in the dynamic and kinematic equations (4.6) and (4.7) is still expressed with respect to the inertial reference frame.

The different choice or the reference frame used to express the state variables has been made only to simplify the exposition of the design procedure. The residual filters practically implemented in Simulink are all exploiting the inertial sensor measurements and the assumed available measure of the satellite position Ω_o along the orbit path given by Eq. (2.57) to derive the orbital attitude Euler parameters for the computation of the external disturbance torques.

This is done since the models of the external disturbance torques in Eq. (2.46) are based on the values of the attitude variables with respect to the orbital reference frame. Hence, both the dynamic and kinematic equations of the spacecraft are written as functions of the orbital attitude of the satellite. The associated fault inputs and the designed filter models are essentially the same, apart the explicit definition of the orbital attitude variables as functions of the inertial ones, but in this way, the relations are simpler to be written.

Finally, it is worth noting that this done only for the description of the FDI module. Instead, the design and description of the FE module, AFTC scheme and SMC controller in the following chapters will exploit the actual sensor measurements of attitude and angular velocity with respect to the inertial reference frame.

Exploiting the modelling procedure described in Section 3.5 and proposed by Mattone and De Luca (2006b) for actuator and sensor faults, respectively, the following fault inputs can be defined for all the considered actuator and sensor faults.

Considering the actuator faults, the physical and corresponding associated mathematical fault inputs are shown. The possible physical faults affecting the four reaction wheels of the ACS and the corresponding vector fields are defined as follows:

$$\begin{aligned}
f_{u_1} &= u_1 - u_{c1} & l_{u_1} &= \begin{bmatrix} -1 & 0 & 0 & 0 & \sqrt{\frac{1}{3}}\frac{1}{I_{xx}} & \sqrt{\frac{2}{3}}\frac{1}{I_{yy}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{u_2} &= u_2 - u_{c2} & l_{u_2} &= \begin{bmatrix} 0 & -1 & 0 & 0 & \sqrt{\frac{1}{3}}\frac{1}{I_{xx}} & -\sqrt{\frac{2}{3}}\frac{1}{I_{yy}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{u_3} &= u_3 - u_{c3} & l_{u_3} &= \begin{bmatrix} 0 & 0 & -1 & 0 & -\sqrt{\frac{1}{3}}\frac{1}{I_{xx}} & 0 & -\sqrt{\frac{2}{3}}\frac{1}{I_{zz}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{u_4} &= u_4 - u_{c4} & l_{u_4} &= \begin{bmatrix} 0 & 0 & 0 & -1 & -\sqrt{\frac{1}{3}}\frac{1}{I_{xx}} & 0 & \sqrt{\frac{2}{3}}\frac{1}{I_{zz}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.9}$$

The possible physical and associated mathematical fault inputs, with the relative number of associated faults and the corresponding vector fields, for the sensors of the ACS measuring the spin rates of the actuators flywheels are defined, for each actuator subsystem, as follows:

1. For the first reaction wheel sensor fault $F_{y_1} = y_1 - x_1$ and $\nu_1 = 4$

$$\begin{aligned}
f_{y_1,1} &= \sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}} x_7 (y_1 - x_1) & l_{x_1,1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_1,2} &= -\sqrt{\frac{1}{3}}\frac{I_w}{I_{yy}} x_7 (y_1 - x_1) & l_{x_1,2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_1,3} &= \sqrt{\frac{1}{3}}\frac{I_w}{I_{zz}} (x_6 - \sqrt{2}x_5)(y_1 - x_1) & l_{x_1,3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_1,4} &= \dot{y}_1 - \dot{x}_1 & l_{x_1,4} &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.10}$$

2. For the second reaction wheel sensor fault $F_{y_2} = y_2 - x_2$ and $\nu_2 = 4$

$$\begin{aligned}
f_{y_2,1} &= -\sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}} x_7 (y_2 - x_2) & l_{x_2,1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_2,2} &= -\sqrt{\frac{1}{3}}\frac{I_w}{I_{yy}} x_7 (y_2 - x_2) & l_{x_2,2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_2,3} &= \sqrt{\frac{1}{3}}\frac{I_w}{I_{zz}} (x_6 + \sqrt{2}x_5)(y_2 - x_2) & l_{x_2,3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_2,4} &= \dot{y}_2 - \dot{x}_2 & l_{x_2,4} &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.11}$$

3. For the third reaction wheel sensor fault $F_{y_3} = y_3 - x_3$ and $\nu_3 = 4$

$$
\begin{aligned}
f_{y_3,1} &= \sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}} x_6 (y_3 - x_3) & l_{x_3,1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_3,2} &= \sqrt{\frac{1}{3}}\frac{I_w}{I_{yy}} (x_7 - \sqrt{2}x_5)(y_3 - x_3) & l_{x_3,2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_3,3} &= -\sqrt{\frac{1}{3}}\frac{I_w}{I_{zz}} x_6 (y_3 - x_3) & l_{x_3,3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_3,4} &= \dot{y}_3 - \dot{x}_3 & l_{x_3,4} &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
\tag{4.12}
$$

4. For the fourth reaction wheel sensor fault $F_{y_4} = y_4 - x_4$ and $\nu_4 = 4$

$$
\begin{aligned}
f_{y_4,1} &= -\sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}} x_6 (y_4 - x_4) & l_{x_4,1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_4,2} &= \sqrt{\frac{1}{3}}\frac{I_w}{I_{yy}} (x_7 + \sqrt{2}x_5)(y_4 - x_4) & l_{x_4,2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_4,3} &= -\sqrt{\frac{1}{3}}\frac{I_w}{I_{zz}} x_6 (y_4 - x_4) & l_{x_4,3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_4,4} &= \dot{y}_4 - \dot{x}_4 & l_{x_4,4} &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
\tag{4.13}
$$

It is worth noting that in the previous relations the fault inputs $f_{y_k,i}$ associated to the *k-th* physical sensor fault F_{y_k} use a subscript notation with "*y*" instead of "*x*" just to avoid any misinterpretation due to the presence of two redundant star tracker sensors measuring the same quaternion state vector. The possible physical and associated mathematical fault inputs, with the relative number of associated faults and the corresponding vector fields, for the sensors of the ADS measuring the satellite angular velocities with respect to each body axis are defined, for each rate gyro sensor, as follows:

1. For the first spacecraft angular rate sensor fault $F_{y_5} = y_5 - x_5$ and $\nu_5 = 7$

$$
\begin{aligned}
f_{y_5,1} &= \left(-\frac{(I_{xx} - I_{zz})}{I_{yy}} x_7 + \sqrt{\frac{2}{3}}\frac{I_w}{I_{yy}} (x_4 - x_3) \right)(y_5 - x_5) & l_{x_5,1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_5,2} &= \left(-\frac{(I_{yy} - I_{xx})}{I_{zz}} x_6 + \sqrt{\frac{2}{3}}\frac{I_w}{I_{zz}} (x_2 - x_1) \right)(y_5 - x_5) & l_{x_5,2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_5,3} &= x_{11}(y_5 - x_5) & l_{x_5,3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 \end{bmatrix}^T \\
f_{y_5,4} &= x_{10}(y_5 - x_5) & l_{x_5,4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/2 & 0 & 0 \end{bmatrix}^T \\
f_{y_5,5} &= x_9(y_5 - x_5) & l_{x_5,5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 \end{bmatrix}^T \\
f_{y_5,6} &= x_8(y_5 - x_5) & l_{x_5,6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 \end{bmatrix}^T \\
f_{y_5,7} &= \dot{y}_5 - \dot{x}_5 & l_{x_5,7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
\tag{4.14}
$$

2. For the second spacecraft angular rate sensor fault $F_{y_6} = y_6 - x_6$ and $\nu_6 = 7$

$$\begin{aligned}
&f_{y_6,1} = \left(-\frac{(I_{zz}-I_{yy})}{I_{xx}}x_7 + \sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}}(x_3-x_4)\right)(y_6-x_6) && l_{x_6,1} = \begin{bmatrix}0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_6,2} = \left(-\frac{(I_{yy}-I_{xx})}{I_{zz}}x_5 + \sqrt{\frac{1}{3}}\frac{I_w}{I_{zz}}(x_1+x_2-x_3-x_4)\right)(y_6-x_6) && l_{x_6,2} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_6,3} = x_{10}(y_6-x_6) && l_{x_6,3} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_6,4} = x_{11}(y_6-x_6) && l_{x_6,4} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/2 & 0 & 0\end{bmatrix}^T \\
&f_{y_6,5} = x_8(y_6-x_6) && l_{x_6,5} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/2 & 0\end{bmatrix}^T \\
&f_{y_6,6} = x_9(y_6-x_6) && l_{x_6,6} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2\end{bmatrix}^T \\
&f_{y_6,7} = \dot{y}_6 - \dot{x}_6 && l_{x_6,7} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T
\end{aligned} \tag{4.15}$$

3. For the third spacecraft angular rate sensor fault $F_{y_7} = y_7 - x_7$ and $\nu_7 = 7$

$$\begin{aligned}
&f_{y_7,1} = \left(-\frac{(I_{zz}-I_{yy})}{I_{xx}}x_6 + \sqrt{\frac{2}{3}}\frac{I_w}{I_{xx}}(x_1-x_2)\right)(y_7-x_7) && l_{x_7,1} = \begin{bmatrix}0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_7,2} = \left(-\frac{(I_{xx}-I_{zz})}{I_{yy}}x_5 + \sqrt{\frac{1}{3}}\frac{I_w}{I_{yy}}(x_3+x_4-x_1-x_2)\right)(y_7-x_7) && l_{x_7,2} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_7,3} = x_9(y_7-x_7) && l_{x_7,3} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0\end{bmatrix}^T \\
&f_{y_7,4} = x_8(y_7-x_7) && l_{x_7,4} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0\end{bmatrix}^T \\
&f_{y_7,5} = x_{11}(y_7-x_7) && l_{x_7,5} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/2 & 0\end{bmatrix}^T \\
&f_{y_7,6} = x_{10}(y_7-x_7) && l_{x_7,6} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2\end{bmatrix}^T \\
&f_{y_7,7} = \dot{y}_7 - \dot{x}_7 && l_{x_7,7} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T
\end{aligned} \tag{4.16}$$

Finally, the possible physical and associated mathematical fault inputs, with the relative number of associated faults and the corresponding vector fields, for the sensors of the ADS measuring the satellite attitude in quaternion notation are defined as follows. In this case, since two distinct and redundant attitude sensors are assumed to be present, two distinct sets of mathematical faults are defined, one for each sensor output vector $\bar{\mathbf{q}}_{star1} = [q_{star1,1}\ q_{star1,2}\ q_{star1,3}\ q_{star1,4}]^T_{measured}$ and $\bar{\mathbf{q}}_{star2} = [q_{star2,1}\ q_{star2,2}\ q_{star2,3}\ q_{star2,4}]^T_{measured}$, with respect to the same quaternion state vector $\bar{\mathbf{q}}_o = \begin{bmatrix}q_{1,o} & q_{2,o} & q_{3,o} & q_{4,o}\end{bmatrix}^T$. Moreover, associated fault inputs are defined for each component of the quaternion vector.

For the first attitude sensor, it can be defined:

1. the fault $F_{y_8} = y_8 - x_8$ on the first quaternion parameter with $\nu_8 = 7$

$$
\begin{aligned}
f_{y_8,1} &= \frac{6\mu}{R^3}\frac{(I_{zz}-I_{yy})}{I_{xx}}\left(-2x_{11}(y_8^3-x_8^3)-2x_9x_{10}(y_8^2-x_8^2)+(x_{11}-2x_9^2x_{11})(y_8-x_8)\right)+2\frac{F_{aero}}{I_{xx}}(x_{10}c_{p,y}-x_9c_{p,z})(y_8-x_8)\\
f_{y_8,2} &= \frac{6\mu}{R^3}\frac{(I_{xx}-I_{zz})}{I_{yy}}\left(-2x_{10}(y_8^3-x_8^3)+2x_9x_{11}(y_8^2-x_8^2)+(x_{10}-2x_9^2x_{10})(y_8-x_8)\right)-2\frac{F_{aero}}{I_{yy}}x_{10}c_{p,x}(y_8-x_8)\\
f_{y_8,3} &= \frac{12\mu}{R^3}\frac{(I_{yy}-I_{xx})}{I_{zz}}\left(2x_{10}x_{11}(y_8^2-x_8^2)+(x_9x_{10}^2-x_9x_{11}^2)(y_8-x_8)\right)+2\frac{F_{aero}}{I_{zz}}x_9c_{p,x}(y_8-x_8)\\
f_{y_8,4} &= x_7(y_8-x_8)\\
f_{y_8,5} &= (x_6-\omega_o)(y_8-x_8)\\
f_{y_8,6} &= x_5(y_8-x_8)\\
f_{y_8,7} &= \dot{y}_8-\dot{x}_8
\end{aligned}
\tag{4.17}
$$

2. The fault $F_{y_9} = y_9 - x_9$ on the second quaternion parameter with $\nu_9 = 7$

$$
\begin{aligned}
f_{y_9,1} &= \frac{6\mu}{R^3}\frac{(I_{zz}-I_{yy})}{I_{xx}}\left(-2x_{10}(y_9^3-x_9^3)-2x_8x_{11}(y_9^2-x_9^2)+(x_{10}-2x_8^2x_{10})(y_9-x_9)\right)+2\frac{F_{aero}}{I_{xx}}(x_{11}c_{p,y}-x_8c_{p,z})(y_9-x_9)\\
f_{y_9,2} &= \frac{6\mu}{R^3}\frac{(I_{xx}-I_{zz})}{I_{yy}}\left(2x_{11}(y_9^3-x_9^3)-2x_8x_{10}(y_9^2-x_9^2)+(2x_8^2x_{11}-x_{11})(y_9-x_9)\right)-2\frac{F_{aero}}{I_{yy}}\left(c_{p,z}(y_9^2-x_9^2)+x_{11}c_{p,x}(y_9-x_9)\right)\\
f_{y_9,3} &= \frac{12\mu}{R^3}\frac{(I_{yy}-I_{xx})}{I_{zz}}\left(-x_{10}x_{11}(y_9^2-x_9^2)+x_8x_{10}^2(y_9-x_9)\right)+2\frac{F_{aero}}{I_{zz}}\left(c_{p,y}(y_9^2-x_9^2)+x_8c_{p,x}(y_9-x_9)\right)\\
f_{y_9,4} &= x_7(y_9-x_9)\\
f_{y_9,5} &= x_5(y_9-x_9)\\
f_{y_9,6} &= (x_6+\omega_o)(y_9-x_9)\\
f_{y_9,7} &= \dot{y}_9-\dot{x}_9
\end{aligned}
\tag{4.18}
$$

3. The fault $F_{y_{10}} = y_{10} - x_{10}$ on the third quaternion parameter with $\nu_{10} = 7$

$$
\begin{aligned}
f_{y_{10},1} &= \frac{6\mu}{R^3}\frac{(I_{zz}-I_{yy})}{I_{xx}}(x_9-2x_8^2x_9-2x_9^3)(y_{10}-x_{10})+2\frac{F_{aero}}{I_{xx}}(x_8c_{p,y}+x_{11}c_{p,z})(y_{10}-x_{10})\\
f_{y_{10},2} &= \frac{6\mu}{R^3}\frac{(I_{xx}-I_{zz})}{I_{yy}}(x_8-2x_8^3-2x_8x_9^2)(y_{10}-x_{10})-2\frac{F_{aero}}{I_{yy}}\left(c_{p,z}(y_{10}^2-x_{10}^2)+x_8c_{p,x}(y_{10}-x_{10})\right)\\
f_{y_{10},3} &= \frac{12\mu}{R^3}\frac{(I_{yy}-I_{xx})}{I_{zz}}\left(x_8x_9(y_{10}^2-x_{10}^2)+(x_8^2x_{11}-x_9^2x_{11})(y_{10}-x_{10})\right)+2\frac{F_{aero}}{I_{zz}}\left(c_{p,y}(y_{10}^2-x_{10}^2)-x_{11}c_{p,x}(y_{10}-x_{10})\right)\\
f_{y_{10},4} &= (x_6-\omega_o)(y_{10}-x_{10})\\
f_{y_{10},5} &= x_5(y_{10}-x_{10})\\
f_{y_{10},6} &= x_7(y_{10}-x_{10})\\
f_{y_{10},7} &= \dot{y}_{10}-\dot{x}_{10}
\end{aligned}
\tag{4.19}
$$

4. The fault $F_{y_{11}} = y_{11} - x_{11}$ on the fourth quaternion parameter with $\nu_{11} = 7$

$$\begin{aligned}
f_{y_{11},1} &= \frac{6\mu}{R^3}\frac{(I_{zz}-I_{yy})}{I_{xx}}(x_8 - 2x_8^3 - 2x_9^2 x_8)(y_{11}-x_{11}) + 2\frac{F_{aero}}{I_{xx}}(x_9 c_{p,y} + x_{10} c_{p,z})(y_{11}-x_{11}) \\
f_{y_{11},2} &= \frac{6\mu}{R^3}\frac{(I_{xx}-I_{zz})}{I_{yy}}(-x_9 + 2x_8^2 x_9 + 2x_9^3)(y_{11}-x_{11}) - 2\frac{F_{aero}}{I_{yy}} x_9 c_{p,x}(y_{11}-x_{11}) \\
f_{y_{11},3} &= \frac{12\mu}{R^3}\frac{(I_{yy}-I_{xx})}{I_{zz}}\left(-x_8^2(y_{11}^2 - x_{11}^2) + (x_8^2 x_{10} - x_9^2 x_{10})(y_{11}-x_{11})\right) - 2\frac{F_{aero}}{I_{zz}} x_{10} c_{p,x}(y_{11}-x_{11}) \\
f_{y_{11},4} &= x_5(y_{11}-x_{11}) \\
f_{y_{11},5} &= (x_6 + \omega_o)(y_{11}-x_{11}) \\
f_{y_{11},6} &= x_7(y_{11}-x_{11}) \\
f_{y_{11},7} &= \dot{y}_{11} - \dot{x}_{11}
\end{aligned} \tag{4.20}$$

with the corresponding vector fields of the associated faults for the four quaternion components defined as:

$$\begin{aligned}
l_{x_8,1} &= \begin{bmatrix}0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,2} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,3} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,4} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,5} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,6} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_8,7} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T
\end{aligned} \tag{4.21}$$

$$\begin{aligned}
l_{x_9,1} &= \begin{bmatrix}0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,2} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,3} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,4} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,5} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,6} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0\end{bmatrix}^T \\
l_{x_9,7} &= \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T
\end{aligned} \tag{4.22}$$

$$
\begin{aligned}
l_{x_{10},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{10},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
\tag{4.23}
$$

$$
\begin{aligned}
l_{x_{11},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{11},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
\tag{4.24}
$$

Similarly, for the second attitude sensor:

1. The fault $F_{y_{12}} = y_{12} - x_8$ on the first quaternion parameter with $\nu_{12} = 7$

$$
\begin{aligned}
f_{y_{12},1} &= \frac{6\mu}{R^3}\frac{(I_{zz}-I_{yy})}{I_{xx}}\left(-2x_{11}(y_{12}^3 - x_8^3) - 2x_9x_{10}(y_{12}^2 - x_8^2) + (x_{11} - 2x_9^2x_{11})(y_{12} - x_8)\right) + 2\frac{F_{aero}}{I_{xx}}(x_{10}c_{p,y} - x_9c_{p,z})(y_{12} - x_8) \\
f_{y_{12},2} &= \frac{6\mu}{R^3}\frac{(I_{xx}-I_{zz})}{I_{yy}}\left(-2x_{10}(y_{12}^3 - x_8^3) + 2x_9x_{11}(y_{12}^2 - x_8^2) + (x_{10} - 2x_9^2x_{10})(y_{12} - x_8)\right) - 2\frac{F_{aero}}{I_{yy}}x_{10}c_{p,x}(y_{12} - x_8) \\
f_{y_{12},3} &= \frac{12\mu}{R^3}\frac{(I_{yy}-I_{xx})}{I_{zz}}\left(2x_{10}x_{11}(y_{12}^2 - x_8^2) + (x_9x_{10}^2 - x_9x_{11}^2)(y_{12} - x_8)\right) + 2\frac{F_{aero}}{I_{zz}}x_9c_{p,x}(y_{12} - x_8) \\
f_{y_{12},4} &= x_7(y_{12} - x_8) \\
f_{y_{12},5} &= (x_6 - \omega_o)(y_{12} - x_8) \\
f_{y_{12},6} &= x_5(y_{12} - x_8) \\
f_{y_{12},7} &= \dot{y}_{12} - \dot{x}_8
\end{aligned}
\tag{4.25}
$$

2. The fault $F_{y_{13}} = y_{13} - x_{13}$ on the second quaternion parameter with $\nu_{13} = 7$

$$\begin{aligned}
f_{y_{13},1} &= \frac{6\mu}{R^3}\frac{(I_{zz} - I_{yy})}{I_{xx}}\left(-2x_{10}(y_{13}^3 - x_9^3) - 2x_8 x_{11}(y_{13}^2 - x_9^2) + (x_{10} - 2x_8^2 x_{10})(y_{13} - x_9)\right) + 2\frac{F_{aero}}{I_{xx}}(x_{11}c_{p,y} - x_8 c_{p,z})(y_{13} - x_9) \\
f_{y_{13},2} &= \frac{6\mu}{R^3}\frac{(I_{xx} - I_{zz})}{I_{yy}}\left(2x_{11}(y_{13}^3 - x_9^3) - 2x_8 x_{10}(y_{13}^2 - x_9^2) + (2x_8^2 x_{11} - x_{11})(y_{13} - x_9)\right) - 2\frac{F_{aero}}{I_{yy}}\left(c_{p,z}(y_{13}^2 - x_9^2) + x_{11}c_{p,x}(y_{13} - x_9)\right) \\
f_{y_{13},3} &= \frac{12\mu}{R^3}\frac{(I_{yy} - I_{xx})}{I_{zz}}\left(-x_{10}x_{11}(y_{13}^2 - x_9^2) + x_8 x_{10}^2(y_{13} - x_9)\right) + 2\frac{F_{aero}}{I_{zz}}\left(c_{p,y}(y_{13}^2 - x_9^2) + x_8 c_{p,x}(y_{13} - x_9)\right) \\
f_{y_{13},4} &= x_7(y_{13} - x_9) \\
f_{y_{13},5} &= x_5(y_{13} - x_9) \\
f_{y_{13},6} &= (x_6 + \omega_o)(y_{13} - x_9) \\
f_{y_{13},7} &= \dot{y}_{13} - \dot{x}_9
\end{aligned} \tag{4.26}$$

3. The fault $F_{y_{14}} = y_{14} - x_{10}$ on the third quaternion parameter with $\nu_{14} = 7$

$$\begin{aligned}
f_{y_{14},1} &= \frac{6\mu}{R^3}\frac{(I_{zz} - I_{yy})}{I_{xx}}(x_9 - 2x_8^2 x_9 - 2x_9^3)(y_{14} - x_{10}) + 2\frac{F_{aero}}{I_{xx}}(x_8 c_{p,y} + x_{11}c_{p,z})(y_{14} - x_{10}) \\
f_{y_{14},2} &= \frac{6\mu}{R^3}\frac{(I_{xx} - I_{zz})}{I_{yy}}(x_8 - 2x_8^3 - 2x_8 x_9^2)(y_{14} - x_{10}) - 2\frac{F_{aero}}{I_{yy}}\left(c_{p,z}(y_{14}^2 - x_{10}^2) + x_8 c_{p,x}(y_{14} - x_{10})\right) \\
f_{y_{14},3} &= \frac{12\mu}{R^3}\frac{(I_{yy} - I_{xx})}{I_{zz}}\left(x_8 x_9(y_{14}^2 - x_{10}^2) + (x_8^2 x_{11} - x_9^2 x_{11})(y_{14} - x_{10})\right) + 2\frac{F_{aero}}{I_{zz}}\left(c_{p,y}(y_{14}^2 - x_{10}^2) - x_{11}c_{p,x}(y_{14} - x_{10})\right) \\
f_{y_{14},4} &= (x_6 - \omega_o)(y_{14} - x_{10}) \\
f_{y_{14},5} &= x_5(y_{14} - x_{10}) \\
f_{y_{14},6} &= x_7(y_{14} - x_{10}) \\
f_{y_{14},7} &= \dot{y}_{14} - \dot{x}_{10}
\end{aligned} \tag{4.27}$$

4. The fault $F_{y_{15}} = y_{15} - x_{11}$ on the first quaternion component with $\nu_{15} = 7$

$$\begin{aligned}
f_{y_{15},1} &= \frac{6\mu}{R^3}\frac{(I_{zz} - I_{yy})}{I_{xx}}(x_8 - 2x_8^3 - 2x_9^2 x_8)(y_{11} - x_{11}) + 2\frac{F_{aero}}{I_{xx}}(x_9 c_{p,y} + x_{10}c_{p,z})(y_{15} - x_{11}) \\
f_{y_{15},2} &= \frac{6\mu}{R^3}\frac{(I_{xx} - I_{zz})}{I_{yy}}(-x_9 + 2x_8^2 x_9 + 2x_9^3)(y_{15} - x_{11}) - 2\frac{F_{aero}}{I_{yy}}x_9 c_{p,x}(y_{15} - x_{11}) \\
f_{y_{15},3} &= \frac{12\mu}{R^3}\frac{(I_{yy} - I_{xx})}{I_{zz}}\left(-x_8^2(y_{15}^2 - x_{11}^2) + (x_8^2 x_{10} - x_9^2 x_{10})(y_{15} - x_{11})\right) - 2\frac{F_{aero}}{I_{zz}}x_{10}c_{p,x}(y_{15} - x_{11}) \\
f_{y_{15},4} &= x_5(y_{15} - x_{11}) \\
f_{y_{15},5} &= (x_6 + \omega_o)(y_{15} - x_{11}) \\
f_{y_{15},6} &= x_7(y_{15} - x_{11}) \\
f_{y_{15},7} &= \dot{y}_{15} - \dot{x}_{11}
\end{aligned} \tag{4.28}$$

with the corresponding vector fields of the associated faults for the four quaternion components defined as

$$\begin{aligned}
l_{x_{12},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{12},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.29}$$

$$\begin{aligned}
l_{x_{13},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{13},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.30}$$

$$\begin{aligned}
l_{x_{14},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{14},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.31}$$

$$\begin{aligned}
l_{x_{15},1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},2} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},3} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},4} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},5} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},6} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
l_{x_{15},7} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned} \tag{4.32}$$

Finally, since the gravitational disturbance model is almost perfectly known, the FDI robustness is achieved by exploiting an explicit disturbance decoupling based on the NLGA, applied only to the aerodynamic force term. This term represents the main source of uncertainty in the satellite dynamic model, mainly due to the lack of knowledge of the accurate values of air density and satellite drag coefficient. By considering the external aerodynamic disturbance as an additional and always concurrent fault input to be decoupled, with the corresponding vector field

$$d = F_{aero} = -\frac{1}{2}\rho V^2 C_D S_{\text{exposed}} \qquad p(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dfrac{(2x_8x_{10}c_{p,y} + 2x_9x_{11}c_{p,y} - 2x_8x_9c_{p,z} + 2x_{10}x_{11}c_{p,z})}{I_{xx}} \\ \dfrac{(c_{p,z} - 2x_9^2c_{p,z} - 2x_{10}^2c_{p,z} - 2x_8x_{10}c_{p,x} - 2x_9x_{11}c_{p,x})}{I_{yy}} \\ \dfrac{(2x_8x_9c_{p,x} - 2x_{10}x_{11}c_{p,x} - c_{p,y} + 2x_9^2c_{p,y} + 2x_{10}^2c_{p,y})}{I_{zz}} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.33}$$

a nonlinear model affine with respect to all the control inputs, fault inputs and disturbance can be defined, with the associated fault inputs defined above, in the form

$$\dot{y} = g_0(y) + \sum_{i=1}^{m} g_i(y)u_{c,i} + \sum_{k=1}^{s} l_{u_i}(y)\, f_{u_i} + \sum_{l=1}^{p}\sum_{j=1}^{\nu_k} l_{x_{l,j}}(y)\, f_{x_{l,j}} + p(y)d \tag{4.34}$$

and used in the incremental procedure described in Section 3.7 for the design of residual filters decoupled from the aerodynamic disturbance. Any discrepancy between the faultless and faulted system dynamics is fully summarized by means of the introduced mathematical fault inputs.

4.2 Design of Aerodynamic Decoupled Residual Generators for FDI

Considering the new affine model of Eq. (4.34) determined by exploiting the actuator and sensor fault modelling procedure, by following the incremental procedure illustrated in Section 3.7, it is possible to design a set of disturbance decoupled scalar NLGA residual filters and organise them in four banks of filters working in parallel. These banks of residual filters are referred to as RB1 (also defined as local filters), RB2 (also defined as global filters), RB3 and RB4.

For different subsets of mathematical fault inputs it is possible to determine new scalar $\bar{x}_{1s}$-subsystems, with each subsystem affected only by a specific subset of mathematical fault inputs and decoupled from all the other mathematical fault inputs and the aerodynamic disturbance. Each of these newly defined scalar $\bar{x}_{1s}$-subsystems can be modelled as follows:

$$\dot{\bar{y}}_{1s} = n_{1s}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) + \sum_{i=1}^{m} g_{1s,i}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) u_{c,i} + \sum_{k=1}^{s} l_{1s,u_i}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) f_{u_i} + \sum_{l=1}^{p} \sum_{j=1}^{v_k} l_{1s,x_{l,j}}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) f_{x_{l,j}} \quad (4.35)$$

and the corresponding scalar residual generators, written in filter form (Benini *et al.* 2008), are modelled as follows:

$$\begin{cases} \dot{\xi} = n_{1s}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) + \sum_{i=1}^{m} g_{1s,i}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) u_{c,i} + K(\bar{y}_{1s} - \xi) \\ r = \bar{y}_{1s} - \xi \end{cases} \quad (4.36)$$

where the residual signal $r = \bar{y}_{1s} - \xi$ is characterized by the following dynamics:

$$\dot{r} = -K(\bar{y}_{1s} - \xi) + \sum_{k=1}^{s} l_{1s,u_i}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) f_{u_i} + \sum_{l=1}^{p} \sum_{j=1}^{v_k} l_{1s,x_{l,j}}(\bar{y}_{1s}, \bar{y}_{1c}, \bar{y}_2) f_{x_{l,j}} \quad (4.37)$$

Implementing the design procedure described in Section 3.7 for the definition of the set R of residuals, it results that the residual matrix reported in Tab. 4.1 can be determined. This residual matrix describes the sensitivity of each of the designed residuals to all the possible physical faults that could affect either the attitude control inputs or the flywheel spin rate sensor measurements, spacecraft attitude and angular velocity sensor measurements. Each element of this matrix is equal to 1 if the residual is sensitive to the physical fault, or at least to one mathematical fault input associate with it, and equal to 0 is the residual is not sensitive to the physical fault or to any associated mathematical fault input.

Residuals	RB1				RB2					RB3									RB4								
Physical faults and aerodynamic disturbance	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	r_{11}	r_{12}	r_{13}	r_{14}	r_{15}	r_{16}	r_{17}	r_{18}	r_{19}	r_{20}	r_{21}	r_{22}	r_{23}	r_{24}	r_{25}	r_{26}	r_{27}
$F_{u1}=f_{Tctr,1}$	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{u2}=f_{Tctrl,2}$	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{u3}=f_{Tctrl,3}$	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{u4}=f_{Tctrl,4}$	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{y1}=f_{\omega w,1}$	1	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{y2}=f_{\omega w,2}$	0	1	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{y3}=f_{\omega w,3}$	0	0	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{y4}=f_{\omega w,4}$	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$F_{y5}=f_{\omega,x}$	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	0	0	0
$F_{y6}=f_{\omega,y}$	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1
$F_{y7}=f_{\omega,z}$	0	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1
$F_{y8}=f_{\bar{q}star1,1}$	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
$F_{y9}=f_{\bar{q}star1,2}$	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
$F_{y10}=f_{\bar{q}star1,3}$	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
$F_{y11}=f_{\bar{q}star1,4}$	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
$F_{y12}=f_{\bar{q}star2,1}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
$F_{y13}=f_{\bar{q}star2,2}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
$F_{y14}=f_{\bar{q}star2,3}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
$F_{y14}=f_{\bar{q}star2,4}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
$d=F_{aero}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 4.1 - Residual matrix/isolation logic from physical faults to residuals.

Moreover, it is worth noting that all the defined residuals results to be decoupled from the external aerodynamic disturbance term. From the last row of the residual matrix, it is clear that all the obtained residuals result to be explicitly decoupled from the external aerodynamic disturbance, thus allowing to avoid any risk of misleading fault detection and isolation due to uncertainties in the knowledge of the actual values of the disturbance parameters to be used in the model-based residual filter. The external aerodynamic disturbance is considered as an additional fault input from which all the designed residuals have to be decoupled.
In particular, the following four sets of new scalar variables can be determined by using the described procedure, each of them sensitive only to a specific set of fault inputs, leading to the design of the corresponding banks of aerodynamic decoupled residual filters for FDI, organized in generalized schemes:

Residual Bank RB1:
The model equations of the four designed local residual filters composing RB1 result to be sensitive only to possible faults affecting the actuated control torques and the flywheel spin rate sensors. Each residual filter is designed for a specific actuator subsystem, and the generated diagnostic signal can be used to detect possible faults affecting either the actuated torque or the corresponding flywheel sensor measurement of the subsystem. The filters rely only on local measurements of the spin rates of the actuator flywheels. However, the exploitation of only these local residual filters does not permit the complete fault isolation, but only the isolation of the faulty actuator subsystem. It is not possible to distinguish the actual type of the occurred fault. The exploitation of the NLGA is not actually required, *i.e.* its application is straightforward, since each of the original dynamic equations (4.5) of the actuators, on which these filters are based, results to be already independent from any external disturbance and dedicated to a specific actuator subsystem. NLGA does not result to be practically exploitable for the complete isolation of specific faults in the actuator subsystems because there is no way to define coordinate transformations allowing to obtain new quotient subsystems which are unaffected by all faults but one. The four local scalar residual filters are:

1. The residual filter designed from the scalar variable $\bar{y}_{1s,1} = \omega_{w,1}$, with the residual $r_1 = \bar{y}_{1s,1} - \xi_1 = \omega_{w,1} - \xi_1$ sensitive only to the two possible faults $f_{u_1} = u_1 - u_{c1} = T_{ctrl,1} - T_{c,1}$ and $f_{x_1,4} = \dot{y}_1 - \dot{x}_1 = \dot{\omega}_{w,1_{measured}} - \dot{\omega}_{w,1_{true}}$:

$$\begin{cases} \dot{\xi}_1 = -\dfrac{u_1}{I_w} + K_1 r_1 \\ r_1 = \omega_{w,1} - \xi_1 \end{cases} \tag{4.38}$$

2. The residual filter designed from the scalar variable $\bar{y}_{1s,2} = \omega_{w,2}$, with the residual $r_2 = \bar{y}_{1s,2} - \xi_2 = \omega_{w,2} - \xi_2$ sensitive only to the two possible faults $f_{u_2} = u_2 - u_{c2} = T_{ctrl,2} - T_{c,2}$ and $f_{x_2,4} = \dot{y}_2 - \dot{x}_2 = \dot{\omega}_{w,2_{measured}} - \dot{\omega}_{w,2_{true}}$:

$$\begin{cases} \dot{\xi}_2 = -\dfrac{u_2}{I_w} + K_2 r_2 \\ r_2 = \omega_{w,2} - \xi_2 \end{cases} \tag{4.39}$$

3. The residual filter designed from the scalar variable $\bar{y}_{1s,3} = \omega_{w,3}$, with the residual $r_3 = \bar{y}_{1s,3} - \xi_3 = \omega_{w,3} - \xi_3$ sensitive only to the two possible faults $f_{u_3} = u_3 - u_{c3} = T_{ctrl,3} - T_{c,3}$ and $f_{x_3,4} = \dot{y}_3 - \dot{x}_3 = \dot{\omega}_{w,3_{measured}} - \dot{\omega}_{w,3_{true}}$:

$$\begin{cases} \dot{\xi}_3 = -\dfrac{u_3}{I_w} + K_3 r_3 \\ r_3 = \omega_{w,3} - \xi_3 \end{cases} \tag{4.40}$$

4. The residual filter designed from the scalar variable $\bar{y}_{1s,4} = \omega_{w,4}$, with the residual $r_4 = \bar{y}_{1s,4} - \xi_4 = \omega_{w,4} - \xi_4$ sensitive only to the two possible faults $f_{u_4} = u_4 - u_{c4} = T_{ctrl,4} - T_{c,4}$ and $f_{x_4,4} = \dot{y}_4 - \dot{x}_4 = \dot{\omega}_{w,4_{measured}} - \dot{\omega}_{w,4_{true}}$:

$$\begin{cases} \dot{\xi}_4 = -\dfrac{u_4}{I_w} + K_4 r_4 \\ r_4 = \omega_{w,4} - \xi_4 \end{cases} \tag{4.41}$$

The sensitivities of the residuals of RB1 to an occurred physical fault result to be independent from the attitude and manoeuvre condition of the satellite.

Residual Bank RB2:

The global residual filters are based on both the dynamic and kinematic equations (4.6) and (4.7) of the spacecraft model and the dynamic equations (4.5) of the actuators, and thus rely also on global measurements of the spacecraft attitude and angular velocity in addition to the local flywheel spin rate measurements.

The model equations of the designed global residual filters composing RB2 result to be explicitly decoupled from the external aerodynamic disturbance and sensitive, in general, to all the possible faults affecting the flywheel spin rate sensors, spacecraft attitude and angular velocity sensors.

Moreover, thanks to the way the new scalar quotient subsystems have been defined by means of the NLGA, all these global residuals result to be not sensitive also to any fault affecting the actuated control torques. Therefore, once a faulty actuator subsystem has been detected and isolated by means of the residual signals provided by RB1, the complete isolation of the faulty component in the actuator subsystem can be obtained by checking if one or more of the global residual filters provide a diagnostic signal exceeding its selected threshold. The realization of five independent residual filters improves the reliability of the detection and isolation scheme of an occurred physical fault for any attitude or manoeuvre condition or geometric configuration of the spacecraft. These global residual filters are essentially exploited in order to distinguish the actual type of the occurred fault in a faulty actuator subsystem isolated by means of the previously described residual filters of RB1. The exploitation of the NLGA is required since each of the original dynamic equations of the spacecraft, on which these filters are based, presents a torque contribution related to the external aerodynamic disturbance. The need to decouple the external aerodynamic disturbance does not allow to realize residual filters sensitive only to specific sensor or actuator faults, and hence they have been designed to be insensitive also to the control torque faults to realize a more efficient fault isolation. The mathematical models of the five global residual filters can be derived starting from the following scalar variables:

1. The residual filter designed from the scalar variable

$$\begin{aligned} \bar{y}_{1s,5} = {} & (I_{xx}\omega_{x,i} + \sqrt{1/3}\,I_w\omega_{w,1} + \sqrt{1/3}\,I_w\omega_{w,2} - \sqrt{1/3}\,I_w\omega_{w,3} - \sqrt{1/3}\,I_w\omega_{w,4}) \cdot \\ & \cdot(c_{pa,z} - 2q_{star1,2}^2 c_{pa,z} - 2q_{star1,3}^2 c_{pa,z} - 2q_{star1,1}q_{star1,3}c_{pa,x} - 2q_{star1,2}q_{star1,4}c_{pa,x}) + \\ & -(I_{yy}\omega_{y,i} + \sqrt{2/3}\,I_w\omega_{w,1} - \sqrt{2/3}\,I_w\omega_{w,2}) \cdot \\ & \cdot(2q_{star1,1}q_{star1,3}c_{pa,y} + 2q_{star1,2}q_{star1,4}c_{pa,y} - 2q_{star1,1}q_{star1,2}c_{pa,z} + 2q_{star1,3}q_{star1,4}c_{pa,z}) \end{aligned}$$

with the residual $r_5 = \bar{y}_{1s,5} - \xi_5$;

2. The residual filter designed from the scalar variable

$$\begin{aligned}\bar{y}_{1s,6} = &(I_{xx}\omega_{x,i} + \sqrt{1/3}I_w\omega_{w,1} + \sqrt{1/3}I_w\omega_{w,2} - \sqrt{1/3}I_w\omega_{w,3} - \sqrt{1/3}I_w\omega_{w,4})\cdot\\ &\cdot(2q_{star1,1}q_{star1,2}c_{pa,x} - 2q_{star1,3}q_{star1,4}c_{pa,x} - c_{pa,y} + 2q_{star1,2}^2 c_{pa,y} + 2q_{star1,3}^2 c_{pa,y}) +\\ &-(I_{zz}\omega_{z,i} - \sqrt{2/3}I_w\omega_{w,3} + \sqrt{2/3}I_w\omega_{w,4})\cdot\\ &\cdot(2q_{star1,1}q_{star1,3}c_{pa,y} + 2q_{star1,2}q_{star1,4}c_{pa,y} - 2q_{star1,1}q_{star1,2}c_{pa,z} + 2q_{star1,3}q_{star1,4}c_{pa,z})\end{aligned}$$

with the residual $r_6 = \bar{y}_{1s,6} - \xi_6$;

3. The residual filter designed from the scalar variable

$$\begin{aligned}\bar{y}_{1s,7} = &(I_{yy}\omega_{y,i} + \sqrt{2/3}I_w\omega_{w,1} - \sqrt{2/3}I_w\omega_{w,2})\cdot\\ &\cdot(2q_{star1,1}q_{star1,2}c_{pa,x} - 2q_{star1,3}q_{star1,4}c_{pa,x} - c_{pa,y} + 2q_{star1,2}^2 c_{pa,y} + 2q_{star1,3}^2 c_{pa,y}) +\\ &-(I_{zz}\omega_{z,i} - \sqrt{2/3}I_w\omega_{w,3} + \sqrt{2/3}I_w\omega_{w,4})\cdot\\ &\cdot(c_{pa,z} - 2q_{star1,2}^2 c_{pa,z} - 2q_{star1,3}^2 c_{pa,z} - 2q_{star1,1}q_{star1,3}c_{pa,x} - 2q_{star1,2}q_{star1,4}c_{pa,x})\end{aligned}$$

with the residual $r_7 = \bar{y}_{1s,7} - \xi_7$;

4. The residual filter designed from the scalar variable

$$\begin{aligned}\bar{y}_{1s,8} = &(I_{xx}\omega_{x,i} + \sqrt{1/3}I_w\omega_{w,1} + \sqrt{1/3}I_w\omega_{w,2} - \sqrt{1/3}I_w\omega_{w,3} - \sqrt{1/3}I_w\omega_{w,4})\cdot(1 - 2q_{star1,2}^2 - 2q_{star1,3}^2) +\\ &+(I_{yy}\omega_{y,i} + \sqrt{2/3}I_w\omega_{w,1} - \sqrt{2/3}I_w\omega_{w,2})\cdot(2q_{star1,1}q_{star1,2} - 2q_{star1,3}q_{star1,4}) +\\ &+(I_{zz}\omega_{z,i} - \sqrt{2/3}I_w\omega_{w,3} + \sqrt{2/3}I_w\omega_{w,4})\cdot(2q_{star1,1}q_{star1,3} + 2q_{star1,2}q_{star1,4})\end{aligned}$$

with the residual $r_8 = \bar{y}_{1s,8} - \xi_8$;

5. The residual filter designed from the scalar variable

$$\begin{aligned}\bar{y}_{1s,8} = &(I_{xx}\omega_{x,i} + \sqrt{1/3}I_w\omega_{w,1} + \sqrt{1/3}I_w\omega_{w,2} - \sqrt{1/3}I_w\omega_{w,3} - \sqrt{1/3}I_w\omega_{w,4})c_{pa,x} +\\ &+(I_{yy}\omega_{y,i} + \sqrt{2/3}I_w\omega_{w,1} - \sqrt{2/3}I_w\omega_{w,2})c_{pa,y} +\\ &+(I_{zz}\omega_{z,i} - \sqrt{2/3}I_w\omega_{w,3} + \sqrt{2/3}I_w\omega_{w,4})c_{pa,z}\end{aligned}$$

with the residual $r_9 = \bar{y}_{1s,9} - \xi_9$.

These residual filters exploit the measurements $\bar{\mathbf{q}}_{star1} = [q_{star1,1}\ q_{star1,2}\ q_{star1,3}\ q_{star1,4}]_{measured}^T$ provided by the first star tracker sensor. The explicit expressions of the designed residual filter models is in this instance omitted due to the high complexity of these filter models. However, they can be derived starting from the variable reported above. The designed filters result to be sensitive, in general, to all the faults except:

$$
\begin{aligned}
f_{u_1} &= u_1 - u_{c1} = T_{ctrl,1} - T_{c,1} \\
f_{u_2} &= u_2 - u_{c2} = T_{ctrl,2} - T_{c,2} \\
f_{u_3} &= u_3 - u_{c3} = T_{ctrl,3} - T_{c,3} \\
f_{u_4} &= u_4 - u_{c4} = T_{ctrl,4} - T_{c,4}
\end{aligned}
$$

Residual Bank RB3:

The model equations of the designed residual filters composing RB3 result to be sensitive only to possible faults affecting the spacecraft attitude and angular velocity sensors.

In particular, thanks to the exploitation of the NLGA, each designed residual filter results to be sensitive to possible faults affecting the first star tracker sensor and a specific couple of rate gyro sensors. Concerning the attitude sensor faults, it is worth recalling that, in general, a fault occurring in a star tracker can affect all the quaternion vector components. Hence, it is not possible to design residual filters sensitive only to faults affecting a specific quaternion vector component and not sensitive to any fault in the other spacecraft ADCS sensors and quaternion vector components. For this reason, a hardware redundancy for the attitude sensors has been considered, and the exploitation of this redundancy allows the accurate isolation of faults occurred in the ADS of the spacecraft. In fact, it results to be not possible to distinguish between faults on an angular velocity sensor or attitude sensor with only a single attitude measurement available with certainty in all the attitude and manoeuvre conditions.

Anyway, it is worth observing that the exploitation of the double redundancy of the attitude sensors would be insufficient to correctly isolate a faulty attitude sensor directly by comparing the different attitude measurements by means of a voting scheme, but only to detect the occurrence of a possible fault. Therefore, in this thesis, the isolation of all faults is carried out by means of residual generators exploiting analytical redundancies.

Concerning the rate gyro sensors, once a fault has occurred, it is detected and exactly isolated thanks to the generalized organization of the designed residual filters. For each spacecraft rate gyro, three out of nine specific residual filters of RB3 results to be always insensitive to the corresponding rate gyro sensor fault, whereas at least two of the other six residuals results to exceed its selected threshold in any attitude and manoeuvre condition. Since each designed residual filter is sensitive to possible faults occurring on a specific couple of rate gyros, any healthy sensor can be accurately isolated by detecting the three related residual filters whose diagnostic signals simultaneously remain within their selected thresholds. Therefore, the actual faulty sensor is isolated by exclusion. The nine scalar residual filters are:

1. The residual filter designed from the scalar variable $\bar{y}_{1s,10} = q_{star1,1}^2 + q_{star1,2}^2$, with the residual $r_{10} = \bar{y}_{1s,10} - \xi_{10} = q_{star1,1}^2 + q_{star1,2}^2 - \xi_{10}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$
\begin{cases}
\dot{\xi}_{10} = 2\omega_{x,i}(q_{star1,1}q_{star1,4} + q_{star1,2}q_{star1,3}) + \\
\quad +2\omega_{y,i}(q_{star1,2}q_{star1,4} - q_{star1,1}q_{star1,3}) + \\
\quad +2\omega_o(q_{star1,1}q_{star1,3} + q_{star1,2}q_{star1,4}) + K_{10}r_{10} \\
r_{10} = q_{star1,1}^2 + q_{star1,2}^2 - \xi_{10}
\end{cases}
\tag{4.42}
$$

2. The residual filter designed from the scalar variable $\bar{y}_{1s,11} = 2(q_{star1,1}q_{star1,3} + q_{star1,2}q_{star1,4})$, with the residual $r_{11} = \bar{y}_{1s,11} - \xi_{11} = 2(q_{star1,1}q_{star1,3} + q_{star1,2}q_{star1,4}) - \xi_{11}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{11} = 2\omega_{x,i}(q_{star1,3}q_{star1,4} - q_{star1,1}q_{star1,2}) + \\ \quad +\omega_{y,i}(1 - 2q_{star1,2}^2 - 2q_{star1,3}^2) + K_{11}r_{11} \\ r_{11} = 2(q_{star1,1}q_{star1,3} + q_{star1,2}q_{star1,4}) - \xi_{11} \end{cases} \tag{4.43}$$

3. The residual filter designed from the scalar variable $\bar{y}_{1s,12} = 2(q_{star1,1}q_{star1,4} - q_{star1,2}q_{star1,3})$, with the residual $r_{12} = \bar{y}_{1s,12} - \xi_{12} = 2(q_{star1,1}q_{star1,4} - q_{star1,2}q_{star1,3}) - \xi_{12}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{12} = \omega_{x,i}(1 - 2q_{star1,1}^2 - 2q_{star1,3}^2) + \\ \quad -2\omega_{y,i}(q_{star1,2}q_{star1,2} + q_{star1,3}q_{star1,4}) + K_{12}r_{12} \\ r_{12} = 2(q_{star1,1}q_{star1,4} - q_{star1,2}q_{star1,3}) - \xi_{12} \end{cases} \tag{4.44}$$

4. The residual filter designed from the scalar variable $\bar{y}_{1s,13} = q_{star1,1}^2 + q_{star1,3}^2$, with the residual $r_{13} = \bar{y}_{1s,13} - \xi_{13} = q_{star1,1}^2 + q_{star1,3}^2 - \xi_{13}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{13} = 2\omega_{x,i}(q_{star1,1}q_{star1,4} - q_{star1,2}q_{star1,3}) + \\ \quad +2\omega_{z,i}(q_{star1,2}q_{star1,2} + q_{star1,3}q_{star1,4}) + K_{13}r_{13} \\ r_{13} = q_{star1,1}^2 + q_{star1,3}^2 - \xi_{13} \end{cases} \tag{4.45}$$

5. The residual filter designed from the scalar variable $\bar{y}_{1s,14} = 2(q_{star1,1}q_{star1,4} + q_{star1,2}q_{star1,3})$, with the residual $r_{14} = \bar{y}_{1s,14} - \xi_{14} = 2(q_{star1,1}q_{star1,4} + q_{star1,2}q_{star1,3}) - \xi_{14}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{14} = \omega_{x,i}(1 - 2q_{star1,1}^2 - 2q_{star1,2}^2) + \\ \quad +2\omega_{z,i}(q_{star1,2}q_{star1,4} - q_{star1,1}q_{star1,3}) + \\ \quad +2\omega_{o}(q_{star1,3}q_{star1,4} - q_{star1,1}q_{star1,2}) + K_{14}r_{14} \\ r_{14} = 2(q_{star1,1}q_{star1,4} + q_{star1,2}q_{star1,3}) - \xi_{14} \end{cases} \tag{4.46}$$

6. The residual filter designed from the scalar variable $\bar{y}_{1s,15} = 2(q_{star1,1}q_{star1,2} - q_{star1,3}q_{star1,4})$, with the residual $r_{15} = \bar{y}_{1s,15} - \xi_{15} = 2(q_{star1,1}q_{star1,2} - q_{star1,3}q_{star1,4}) - \xi_{15}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{15} = 2\omega_{x,i}(q_{star1,1}q_{star1,3} + q_{star1,2}q_{star1,4}) + \\ \quad -\omega_{z,i}(1 - 2q_{star1,2}^2 - 2q_{star1,3}^2) + \\ \quad +2\omega_{o}(q_{star1,2}q_{star1,3} + q_{star1,1}q_{star1,4}) + K_{15}r_{15} \\ r_{15} = 2(q_{star1,1}q_{star1,2} - q_{star1,3}q_{star1,4}) - \xi_{15} \end{cases} \tag{4.47}$$

7. The residual filter designed from the scalar variable $\bar{y}_{1s,16} = q_{star1,2}^2 + q_{star1,3}^2$, with the residual $r_{16} = \bar{y}_{1s,16} - \xi_{16} = q_{star1,2}^2 + q_{star1,3}^2 - \xi_{16}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{16} = 2\omega_{y,i}(q_{star1,2}q_{star1,4} + q_{star1,1}q_{star1,3}) + \\ \quad +2\omega_{z,i}(q_{star1,3}q_{star1,4} - q_{star1,1}q_{star1,2}) + \\ \quad +2\omega_o(q_{star1,2}q_{star1,4} - q_{star1,1}q_{star1,3}) + K_{16}r_{16} \\ r_{16} = q_{star1,2}^2 + q_{star1,3}^2 - \xi_{16} \end{cases} \tag{4.48}$$

8. The residual filter designed from the scalar variable $\bar{y}_{1s,17} = 2(q_{star1,1}q_{star1,2} + q_{star1,3}q_{star1,4})$, with the residual $r_{17} = \bar{y}_{1s,17} - \xi_{17} = 2(q_{star1,1}q_{star1,2} + q_{star1,3}q_{star1,4}) - \xi_{17}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{17} = 2\omega_{y,i}(q_{star1,1}q_{star1,4} - q_{star1,2}q_{star1,3}) + \\ \quad +\omega_{z,i}(1 - 2q_{star1,1}^2 - 2q_{star1,3}^2) + K_{17}r_{17} \\ r_{17} = 2(q_{star1,1}q_{star1,2} + q_{star1,3}q_{star1,4}) - \xi_{17} \end{cases} \tag{4.49}$$

9. The residual filter designed from the scalar variable $\bar{y}_{1s,18} = 2(q_{star1,1}q_{star1,3} - q_{star1,2}q_{star1,4})$, with the residual $r_{18} = \bar{y}_{1s,18} - \xi_{18} = 2(q_{star1,1}q_{star1,3} - q_{star1,2}q_{star1,4}) - \xi_{18}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$:

$$\begin{cases} \dot{\xi}_{18} = -\omega_{y,i}(1 - 2q_{star1,1}^2 - 2q_{star1,2}^2) + \\ \quad +2\omega_{z,i}(q_{star1,2}q_{star1,3} + q_{star1,1}q_{star1,4}) + \\ \quad -2\omega_o(1 - 2q_{star1,2}^2 - 2q_{star1,3}^2) + K_{18}r_{18} \\ r_{18} = 2(q_{star1,1}q_{star1,3} - q_{star1,2}q_{star1,4}) - \xi_{18} \end{cases} \tag{4.50}$$

It is worth recalling that these residual filters are actually sensitive to the fault vector $\bar{\mathbf{F}}_{\mathbf{q}_{star}} = \bar{\mathbf{q}}_{star} \otimes \bar{\mathbf{q}}_f - \bar{\mathbf{q}}_{star}$ representing the additive form associated to the quaternion fault $\bar{\mathbf{q}}_f$, and not the quaternion fault $\bar{\mathbf{q}}_f$ itself. The fault vector $\bar{\mathbf{F}}_{\mathbf{q}_{star}}$ can be estimated once the faulty attitude sensor has been isolated, by means of fault estimators. Moreover, the estimate of this vector can be directly used for fault accommodation. These residual filters exploit the measurements of $\bar{\mathbf{q}}_{star1} = [q_{star1,1}\ q_{star1,2}\ q_{star1,3}\ q_{star1,4}]^T$ provided by the first star tracker sensor.

Moreover, looking at the model equations of these designed filters, it is possible to verify that if a fault affecting a generic rate gyro sensor occurs, at least one of the six sensitive residual filters will exceed its threshold if the fault size is generally large enough, for any attitude condition. Even if the angular velocity variables are multiplied by functions of the spacecraft attitude, it is impossible that all the attitude depending functions multiplying a specific angular velocity variable are simultaneously equal to or almost zero, thus reducing or nullifying the sensitivity of the generally sensitive residual signals to a specific physical fault affecting the rate gyro sensor measurement.

In fact, the attitude depending functions multiplying the angular velocity variables correspond to the elements of the rotation matrix of Eq. (2.14), which is orthogonal by definition. In fact, recalling the expression of the rotation matrix $\mathbf{R}_o^b(\bar{\mathbf{q}})$ representing the satellite attitude with respect to the orbital reference frame, defined as

$$\mathbf{R}_o^b(\bar{\mathbf{q}}) = \begin{bmatrix} 1-2q_2^2-2q_3^2 & 2(q_1q_2+q_3q_4) & 2(q_1q_3-q_2q_4) \\ 2(q_1q_2-q_3q_4) & 1-2q_1^2-q_3^2 & 2(q_2q_3+q_1q_4) \\ 2(q_1q_3+q_2q_4) & 2(q_2q_3-q_1q_4) & 1-2q_1^2-2q_2^2 \end{bmatrix}$$

and considering for example the angular velocity variable $\omega_{x,i}$, it can be seen that the attitude depending functions multiplying it in the mathematical models of the residual filters of RB3 are the six elements composing the last two rows of $\mathbf{R}_o^b(\bar{\mathbf{q}})$. Since the matrix is orthogonal, at least two elements of these six are nonzero in any attitude condition. Consequently, with the assumption of single fault at a time, at least two of the designed residual signals will present significant deviations from their nominal fault-free conditions.

Therefore, the detection of any fault affecting a generic rate gyro sensor is guaranteed for any attitude condition. The faults affecting the quaternion vector are always detectable, for any attitude or manoeuvre condition, since at least the fault inputs associated to the time derivative of the component of the fault vector are not null when a fault occurs, also in a stationary attitude condition. The complete fault isolation is performed as already described, by cross-checking the nine residual signals and verifying if the diagnostic signals provided by RB4 have different or the same behaviours in order to distinguish the type of an occurred fault between quaternion and angular velocity faults.

Residual Bank RB4:

Similarly to the residual filters of RB3, the model equations of the designed residual filters composing RB4 result to be sensitive only to possible faults affecting the spacecraft attitude and angular velocity sensors. In particular, thanks to the exploitation of the NLGA, each designed residual filter results to be sensitive to possible faults affecting the second star tracker sensor and a specific couple of rate gyro sensors. The set of residual filters of RB4 results to be sensitive to faults affecting the second star tracker sensor, whereas RB3 results to be sensitive to faults affecting the first star tracker sensor. Therefore, the isolation of the faulty star tracker sensor is realized by checking the residual filter set whose diagnostic signals exceed their selected thresholds. With the assumption of a single fault occurring at a time, if the occurred fault affects an attitude sensor, only one or more of the diagnostic signals provided by a single specific set of residual filters between RB3 and RB4 should provide information about the occurred fault. On the other hand, since both RB3 and RB4 are fed with the same spacecraft angular velocity measurements, once a fault occurs in one of the spacecraft rate gyros, the residuals in both RB3 and RB4 have the same behaviour, and realize the fault isolation in the same way simultaneously. The models of these nine scalar residual filters are similar to the ones of RB3, with the only difference that the measurements $\bar{\mathbf{q}}_{star2} = [q_{star2,1}\ q_{star2,2}\ q_{star2,3}\ q_{star2,4}]^T_{measured}$ provided by the second star tracker sensor are exploited, and hence their explicit expressions are omitted for simplicity. The same general considerations about fault sensitivities made for RB3 are valid.

1. The residual filter designed from the scalar variable $\bar{y}_{1s,19} = q_{star2,1}^2 + q_{star2,2}^2$, with the residual $r_{19} = \bar{y}_{1s,19} - \xi_{19}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$;
2. The residual filter designed from the scalar variable $\bar{y}_{1s,20} = 2(q_{star2,1}q_{star2,3} + q_{star2,2}q_{star2,4})$, with the residual $r_{20} = \bar{y}_{1s,20} - \xi_{20}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$;

3. The residual filter designed from the scalar variable $\bar{y}_{1s,21} = 2(q_{star2,1}q_{star2,4} - q_{star2,2}q_{star2,3})$, with the residual $r_{21} = \bar{y}_{1s,21} - \xi_{21}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star1}$;
4. The residual filter designed from the scalar variable $\bar{y}_{1s,22} = q^2_{star2,1} + q^2_{star2,3}$, with the residual $r_{22} = \bar{y}_{1s,22} - \xi_{22}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$;
5. The residual filter designed from the scalar variable $\bar{y}_{1s,23} = 2(q_{star2,1}q_{star2,4} + q_{star2,2}q_{star2,3})$, with the residual $r_{23} = \bar{y}_{1s,23} - \xi_{23}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$;
6. The residual filter designed from the scalar variable $\bar{y}_{1s,24} = 2(q_{star2,1}q_{star2,2} - q_{star2,3}q_{star2,4})$, with the residual $r_{24} = \bar{y}_{1s,24} - \xi_{24}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$;
7. The residual filter designed from the scalar variable $\bar{y}_{1s,25} = q^2_{star2,2} + q^2_{star2,3}$, with the residual $r_{25} = \bar{y}_{1s,25} - \xi_{25}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$;
8. The residual filter designed from the scalar variable $\bar{y}_{1s,26} = 2(q_{star2,1}q_{star2,2} + q_{star2,3}q_{star2,4})$, with the residual $r_{26} = \bar{y}_{1s,26} - \xi_{26}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$;
9. The residual filter designed from the scalar variable $\bar{y}_{1s,27} = 2(q_{star2,1}q_{star2,3} - q_{star2,2}q_{star2,4})$, with the residual $r_{27} = \bar{y}_{1s,27} - \xi_{27}$ sensitive only to the fault inputs associated to the sensors measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$.

Tab. 4.2 summarize the new scalar variables determined by means of the NLGA and used to design the FDI scalar residual filters, as defined in the previous descriptions of the four banks of residual filters.

RB1	
Residual	$\bar{y}_{1s}$ variable
r_1	$\omega_{w,1}$
r_2	$\omega_{w,2}$
r_3	$\omega_{w,3}$
r_4	$\omega_{w,4}$

RB3	
Residual	$\bar{y}_{1s}$ variable
r_{10}	$(q_{star1,1}^2+q_{star1,2}^2)$
r_{11}	$2(q_{star1,1}q_{star1,3}+q_{star1,2}q_{star1,4})$
r_{12}	$2(q_{star1,1}q_{star1,4}-q_{star1,2}q_{star1,3})$
r_{13}	$(q_{star1,1}^2+q_{star1,3}^2)$
r_{14}	$2(q_{star1,1}q_{star1,4}+q_{star1,2}q_{star1,3})$
r_{15}	$2(q_{star1,1}q_{star1,2}-q_{star1,3}q_{star1,4})$
r_{16}	$(q_{star1,2}^2+q_{star1,3}^2)$
r_{17}	$2(q_{star1,1}q_{star1,2}+q_{star1,3}q_{star1,4})$
r_{18}	$2(q_{star1,1}q_{star1,3}-q_{star1,2}q_{star1,4})$

RB4	
Residual	$\bar{y}_{1s}$ variable
r_{19}	$(q_{star2,1}^2+q_{star2,2}^2)$
r_{20}	$2(q_{star2,1}q_{star2,3}+q_{star2,2}q_{star2,4})$
r_{21}	$2(q_{star2,1}q_{star2,4}-q_{star2,2}q_{star2,3})$
r_{22}	$(q_{star2,1}^2+q_{star2,3}^2)$
r_{23}	$2(q_{star2,1}q_{star2,4}+q_{star2,2}q_{star2,3})$
r_{24}	$2(q_{star2,1}q_{star2,2}-q_{star2,3}q_{star2,4})$
r_{25}	$(q_{star2,2}^2+q_{star2,3}^2)$
r_{26}	$2(q_{star2,1}q_{star2,2}+q_{star2,3}q_{star2,4})$
r_{27}	$2(q_{star2,1}q_{star2,3}-q_{star2,2}q_{star2,4})$

RB2	
Residual	$\bar{y}_{1s}$ variable
r_5	$\left(I_{xx}\omega_{x,i}+\sqrt{1/3}I_w\omega_{w,1}+\sqrt{1/3}I_w\omega_{w,2}-\sqrt{1/3}I_w\omega_{w,3}-\sqrt{1/3}I_w\omega_{w,4}\right)\left(c_{p,z}-2c_{p,z}q_{s1,2}^2-2c_{p,z}q_{s1,3}^2-2c_{p,x}q_{s1,1}q_{s1,3}-2c_{p,x}q_{s1,2}q_{s1,4}\right)+$ $-(I_{yy}\omega_{y,i}+\sqrt{2/3}I_w\omega_{w,1}-\sqrt{2/3}I_w\omega_{w,2})(2c_{p,y}q_{s1,1}q_{s1,3}+2c_{p,y}q_{s1,2}q_{star1,4}-2c_{p,z}q_{s1,1}q_{s1,2}+2c_{p,z}q_{s1,3}q_{s1,4})$
r_6	$\left(I_{xx}\omega_{x,i}+\sqrt{1/3}I_w\omega_{w,1}+\sqrt{1/3}I_w\omega_{w,2}-\sqrt{1/3}I_w\omega_{w,3}-\sqrt{1/3}I_w\omega_{w,4}\right)\left(-c_{p,y}+2c_{p,y}q_{s1,2}^2+2c_{p,y}q_{s1,3}^2+2c_{p,x}q_{s1,1}q_{s1,2}-2c_{p,x}q_{s1,3}q_{s1,4}\right)+$ $-(I_{zz}\omega_{z,i}-\sqrt{2/3}I_w\omega_{w,3}+\sqrt{2/3}I_w\omega_{w,4})(2c_{p,y}q_{s1,1}q_{s1,3}+2c_{p,y}q_{s1,2}q_{star1,4}-2c_{p,z}q_{s1,1}q_{s1,2}+2c_{p,z}q_{s1,3}q_{s1,4})$
r_7	$\left(I_{yy}\omega_{y,i}+\sqrt{2/3}I_w\omega_{w,1}-\sqrt{2/3}I_w\omega_{w,2}\right)\left(-c_{p,y}+2c_{p,y}q_{s1,2}^2+2c_{p,y}q_{s1,3}^2+2c_{p,x}q_{s1,1}q_{s1,2}-2c_{p,x}q_{s1,3}q_{s1,4}\right)+$ $-(I_{zz}\omega_{z,i}-\sqrt{2/3}I_w\omega_{w,3}+\sqrt{2/3}I_w\omega_{w,4})(c_{p,z}-2c_{p,z}q_{s1,2}^2-2c_{p,z}q_{s1,3}^2-2c_{p,x}q_{s1,1}q_{s1,3}-2c_{p,x}q_{s1,2}q_{s1,4})$
r_8	$\left(I_{xx}\omega_{x,i}+\sqrt{1/3}I_w\omega_{w,1}+\sqrt{1/3}I_w\omega_{w,2}-\sqrt{1/3}I_w\omega_{w,3}-\sqrt{1/3}I_w\omega_{w,4}\right)\left(1-2q_{s1,2}^2-2q_{s1,3}^2\right)+$ $+\left(I_{yy}\omega_{y,i}+\sqrt{2/3}I_w\omega_{w,1}-\sqrt{2/3}I_w\omega_{w,2}\right)\left(2q_{s1,1}q_{s1,2}-2q_{s1,3}q_{s1,4}\right)+(I_{zz}\omega_{z,i}-\sqrt{2/3}I_w\omega_{w,3}+\sqrt{2/3}I_w\omega_{w,4})(2q_{s1,1}q_{s1,3}+2q_{s1,2}q_{s1,4})$
r_9	$\left(I_{xx}\omega_{x,i}+\sqrt{1/3}I_w\omega_{w,1}+\sqrt{1/3}I_w\omega_{w,2}-\sqrt{1/3}I_w\omega_{w,3}-\sqrt{1/3}I_w\omega_{w,4}\right)c_{p,x}+$ $+\left(I_{yy}\omega_{y,i}+\sqrt{2/3}I_w\omega_{w,1}-\sqrt{2/3}I_w\omega_{w,2}\right)c_{p,y}+(I_{zz}\omega_{z,i}-\sqrt{2/3}I_w\omega_{w,3}+\sqrt{2/3}I_w\omega_{w,4})c_{p,z}$

Table 4.2 - Defined NLGA scalar variables

It is worth noting that RB1 and RB2 are the only banks of residual filters sensitive to any fault occurring on the actuator subsystems, *i.e.* in the ACS of the spacecraft, and RB1 allows the direct detection and isolation of the faulty subsystem. With the assumption of a single fault occurring at a time, the cross-checking of the diagnostic signals provided by these filters is sufficient to detect and isolate any possible fault affecting either the actuated control torques or the flywheel spin rate sensor measurements. The cross-checking also of the diagnostic signals provided by RB3 and RB4 is not required if a fault is detected by RB1. On the other hand, the diagnostic signals provided by RB3 and RB4 results to be sensitive only to possible faults occurring on the spacecraft attitude and angular velocity sensors, *i.e.* in the ADS of the spacecraft. Again, with the assumption of a single fault occurring at a time, the cross-checking of the diagnostic signals provided by these filters is sufficient to detect and isolate any possible fault affecting either the star tracker measurements or the rate gyro measurements. The cross-checking also of the diagnostic signals provided by RB1 and RB2 is not required if a fault is detected by RB3 or RB4. Therefore, the detection and isolation of possible faults occurring in the ACS and ADS can be realized by considering these two couple of residual filter sets separately. However, the simultaneous cross-checking of all the provided diagnostic signals can improve the reliability of the fault detection and isolation scheme and avoid the risk of misleading detection and isolation results.

4.3 Supervisor Decision Logic for Fault Isolation

Obviously, if any residual signal diverges from its fault-free condition, a fault is surely present. A wrong detection and isolation of an occurred fault due to possible parametric uncertainties is prevented since the designed residual filters are decoupled from the aerodynamic disturbance. A more accurate analysis of the residuals is needed in order to correctly isolate the occurred fault.

A proper decision logic is applied to correctly detect and isolate the occurrence of a fault affecting the actuated control torques, the flywheel spin rate sensors or the satellite attitude and angular velocity sensors. The joint use of four banks of residual filters requires the exploitation of a decision logic based on the cross-check of the diagnostic signals provided by all the distinct designed banks.

As already mentioned, each of the four residual filters of RB1 results to be sensitive only to the fault affecting a single specific actuator torque and to the fault affecting the corresponding flywheel spin rate sensor of the same actuator subsystem. These residual filters are based on the dynamic equations (4.5) of the actuators. Hence, each local residual filter of this bank is sensitive to all the possible faults affecting only a specific actuator subsystem of the spacecraft ACS. As a consequence, it is clear that the exploitation of only the local residual filters does not allow to realize a complete fault isolation for the actuator subsystems, since they does not permit to directly distinguish if the occurred fault is affecting the actuator torque or the corresponding actuator sensor measurement.

Therefore, it is necessary to exploit also the five designed global residual filters of RB2 to obtain a complete fault isolation for the actuator subsystems. In fact, these global residual filters result to be sensitive in general to all the mathematical fault inputs associated to the sensors of the spacecraft ACS and ADS, *i.e.* the flywheel spin rate sensors and the spacecraft attitude and angular velocity sensors. These residual filters are based on both the dynamic equations (4.5) of the actuators, and the dynamic and kinematic equations (4.6) and (4.7) of the spacecraft. The exploitation of these two banks of residual filters allows the correct detection and isolation of both the actuator and actuator sensor faults. In case of actuator faults, only the generated local residuals of RB1 present deviations from their fault-free conditions, possibly leading the residuals to exceed selected thresholds. On the other hand, in case of faults on the flywheel spin rate sensors, also the generated global residuals of RB2 present in general deviations from their fault-free conditions, possibly leading the residuals to exceed selected thresholds. Hence, the cross-check of the residuals generated by these two banks of filters working in parallel allows both the detection of a fault occurred in a specific actuator subsystem of the ACS by exploiting the local residuals, and the correct discrimination of the actual type of fault by exploiting also the global residuals.

On the contrary, the residual filters of RB3 and RB4 result to be sensitive only to the mathematical faults associated to the spacecraft attitude and angular velocity sensors. In fact, they are based only on the kinematic equations (4.7) of the spacecraft, which are not depending on the attitude control torques generated by the actuators and the corresponding flywheel spin rates. Hence, these two banks of residual filters can be exploited for the detection and isolation of faults on the sensors of the spacecraft ADS independently from the other two banks of residual filters, with the assumption of a single fault at a time.

As already stated, the design of these two banks is based on the same mathematical models for the nine filters in each of them. Each bank of residual filters is fed by the same set of angular velocity measurements, since no redundancies for the satellite rate gyros are considered, and a different attitude quaternion measurement vector provided by one of the two attitude sensors considered in the ADS. The exploitation of a double redundancy of the attitude sensors is required since it would not be possible to distinguish between faults on an angular velocity sensor or attitude sensor with only a single attitude measurement available. In this way, all the possible spacecraft attitude and angular velocity sensor faults can be detected and properly isolated.

Because of the residual filter design, the residuals of both RB3 and RB4 results to be sensitive in general to the faults affecting the rate gyro measurements. Since the residual filters are designed and organized in a generalized scheme, each designed residual filter results to be sensitive to faults possibly affecting a specific couple of spacecraft rate gyros, and insensitive to the fault possibly affecting the remaining spacecraft rate gyros. Moreover, the nine residual filters in RB3 are in general sensitive also to the mathematical fault inputs associated to the physical fault affecting the sensor measurement of the quaternion vector provided by the first attitude sensor. In the same way, the nine residual filters in RB4 are in general sensitive also to the mathematical fault inputs associated to the physical fault affecting the sensor measurement of the quaternion vector provided by the second attitude sensor.

Hence, only the residuals of RB3 are sensitive to faults affecting the quaternion measurement from the first attitude sensor, whilst only the residuals of RB4 are sensitive to faults affecting the quaternion measurement from the second attitude sensor.

In case of fault in a spacecraft rate gyro, the sensitive residuals of both the RB3 and RB4 are sensitive in the same manner to the occurred fault, and present deviations from their nominal fault-free conditions, possibly leading the residuals to exceed selected thresholds. The not sensitive residuals of both the banks does not present deviations from their fault-free conditions and does not exceed the relative selected thresholds.

On the other hand, in case of faults on the first attitude sensor, only the generated residual signals of RB3 present in general deviations from their fault-free conditions, possibly leading the residuals to exceed selected thresholds. Finally, in case of faults on the second attitude sensor, only the generated residual signals of RB4 present in general deviations from their fault-free conditions, possibly leading the residuals to exceed selected thresholds. Hence, the cross-check of the residuals generated by these two banks of filters working in parallel allows both the detection and the correct discrimination of the actual sensor fault occurred in the ADS.

As already stated, the residual matrix of Tab. 4.1 summarizes the sensitivities of the designed residual signals to each of the physical fault inputs. However, it is worth observing that the sensitivity to any specific physical fault of the designed residual filters in RB2, RB3 and RB4 results to be depending on functions of the quaternion vector after the exploitation of the NLGA to decouple the aerodynamic disturbance and other faults.

Finally, due to the possible presence of measurement noise, proper thresholds have to be selected for each generated residual signal in order to achieve the best performances in term of false alarm rate and missed fault detection rate. In practice, the value of these thresholds have been experimentally selected in the fault-free condition and for each residual signal separately. The selected threshold values guarantee a sufficient margin of safety with respect to the greatest residual values in the fault-free condition. In the performed simulations the thresholds have been selected according to the relation

$$T = \text{const} = \pm \nu \sigma_{res}$$

where σ_{res} is the standard deviation of the residual signal computed in a fault-free condition and ν represents a positive constant, selected equal to $\nu = 6$ in the performed simulations. The following Tab. 4.3 reports the selected values, assumed to be constant during the simulation, such as to keep the fault-free residuals safely under these thresholds.

RB1	
Residual	Threshold
r_1	$17 \cdot 10^{-3}$
r_2	$17 \cdot 10^{-3}$
r_3	$17 \cdot 10^{-3}$
r_4	$17 \cdot 10^{-3}$

RB2	
Residual	Threshold
r_5	$7{,}9 \cdot 10^{-3}$
r_6	$3{,}7 \cdot 10^{-3}$
r_7	$7{,}3 \cdot 10^{-3}$
r_8	$23 \cdot 10^{-3}$
r_9	$8{,}1 \cdot 10^{-3}$

RB3	
Residual	Threshold
r_{10}	$4{,}5 \cdot 10^{-5}$
r_{11}	$4{,}6 \cdot 10^{-5}$
r_{12}	$3 \cdot 10^{-5}$
r_{13}	$3{,}8 \cdot 10^{-5}$
r_{14}	$4 \cdot 10^{-5}$
r_{15}	$1{,}6 \cdot 10^{-5}$
r_{16}	$4{,}3 \cdot 10^{-5}$
r_{17}	$4{,}4 \cdot 10^{-5}$
r_{18}	$2{,}6 \cdot 10^{-5}$

RB4	
Residual	Threshold
r_{19}	$4{,}5 \cdot 10^{-5}$
r_{20}	$4{,}6 \cdot 10^{-5}$
r_{21}	$3 \cdot 10^{-5}$
r_{22}	$3{,}8 \cdot 10^{-5}$
r_{23}	$4 \cdot 10^{-5}$
r_{24}	$1{,}6 \cdot 10^{-5}$
r_{25}	$4{,}3 \cdot 10^{-5}$
r_{26}	$4{,}4 \cdot 10^{-5}$
r_{27}	$2{,}6 \cdot 10^{-5}$

Table 4.3 - Selected threshold values for the designed residuals.

4.4 Simulation Results

In order to show the performances brought by the application of the proposed FDI scheme when applied to the considered satellite attitude control system, the results achieved in the Matlab® and Simulink® environments are shown in the following subsections.
It is assumed that the measurement noises affecting the satellite attitude sensors are modelled by stochastic processes with zero mean, and standard deviation equal to $1.4544 \cdot 10^{-5}$ rad, or 3 arcsec. The noises affecting the satellite rotational speed sensors are modelled by stochastic processes with zero mean, and standard deviation equal to $1.4544 \cdot 10^{-5}$ rad/s, or 3 arcsec/s. Concerning the flywheel spin rate sensors, their noises are modelled by stochastic processes with zero mean, and standard deviation equal to 0.1554 rad/s, or 1.5 rpm.
A terrestrial mission with a circular orbit at an altitude of 350 km and null inclination, with a low Earth equatorial orbit radius $R_{orbit} = 6728.140$ km, the Earth mass of $M_{Earth} = 5.9742 \cdot 10^{24}$ kg, and radius of $R_{Earth} = 6378.140$ km is considered. The satellite's body is modelled as a rectangular parallelepiped whose the principal dimensions (depth, width and height) are $d = 0.6$ m, $w = 2$ m, and $h = 7.5$ m, $c_{cp} = [0.1 \quad 0.15 \quad -0.35]$ m is the aerodynamic torque displacement vector, while the inertia values are $A = 330$ kg/m^2, $B = 280$ kg/m^2, $C = 60$ kg/m$^{2.}$ The reaction wheels maximal torque is set to 1.5 Nm. The atmosphere density is $\rho = \rho_{max} = 6 \cdot 10^{-11} \mathrm{kg/m^3}$, the drag coefficient is $C_D = 2.2$, and the orbital velocity with respect to the atmosphere is $V = 8187.63$ m/s for a prograde orbit (*i.e.* an orbit in the same direction of the Earth rotation), and μ is equal to the Earth's gravitational constant, $\mu = 39.86004418 \cdot 10^{13} \mathrm{m^3/s^2}$. A flywheel moment of inertia J=0.005 kg/m^2 and initial flywheel spin rate values $\boldsymbol{\omega}_{w0} = [-5.235 \quad -5.235 \quad -5.235 \quad -5.235]^T$ rad/s (*i.e.* -50 rpm) for the three considered momentum wheels are assumed. A simulation time of 60 s with a sampling time of 0.1 s is considered.

4.4.1 Considered fault scenarios:

Twelve different fault scenarios are considered, concerning both the actuators and sensors in the ACS and ADS, with a single fault at a time starting at $t_{fault} = 10$ s. These considered scenarios aim to cover some quite general fault conditions, which can appear in real applications. The considered fault scenarios are modelled as additive fault functions as follows:

1. **Fault scenario n.1:** step fault, constant bias on the actuated control torque $T_{ctrl,2}$:

$$\begin{cases} f_{T_{ctrl,2}} = 0 \; Nm & \text{for } t < 10 \; s \\ f_{T_{ctrl,2}} = a_f = -0.4 \; Nm & \text{for } t \geq 10 \; s \end{cases}$$

fault size $a_f = -0.4 \; Nm$

2. **Fault scenario n.2:** sinusoidal fault on the actuated control torque $T_{ctrl,2}$:

$$\begin{cases} f_{T_{ctrl,2}} = 0 \; Nm & \text{for } t < 10 \; s \\ f_{T_{ctrl,2}} = -a_f \sin(2\pi t / T_f) = -0.4 \sin(2\pi \, \mathrm{t}/10) \; Nm & \text{for } t \geq 10 \; s \end{cases}$$

amplitude $a_f = 0.4$ *Nm*, period $T_f = 10$ sec

3. **Fault scenario n.3:** rectangular pulse fault on the actuated control torque $T_{ctrl,2}$:

$$\begin{cases} f_{T_{ctrl,2}} = a_f = -0.4\ Nm & \text{for } t \geq \text{phase delay AND pulse is on} \\ f_{T_{ctrl,2}} = 0\ Nm & \text{otherwise} \end{cases}$$

amplitude $a_f = -0.4\ Nm$, period $T_f = 10$ sec
pulse width $= 50\%$ $(0.5T_f)$, phase delay $= 10$ sec

4. **Fault scenario n.4:** ramp fault on the actuated control torque $T_{ctrl,2}$:

$$\begin{cases} f_{T_{ctrl,2}} = 0\ Nm & \text{for } t < 10\ s \\ f_{T_{ctrl,2}} = a_f t(t - t_f) = -0.4(t - t_f)\ Nm & \text{for } t \geq 10\ s \end{cases}$$

fault slope $a_f = -0.4\ Nm/s$

5. **Fault scenario n.5:** failure of the actuator providing the actuated control torque $T_{ctrl,2}$:

$$\begin{cases} f_{T_{ctrl,2}} = 0\ Nm & \text{for } t < 10\ s \\ f_{T_{ctrl,2}} = -T_{ctrl,2} & \text{for } t \geq 10\ s \end{cases}$$

6. **Fault scenario n.6:** step fault, constant bias on the measured flywheel spin rate $\omega_{w,3}$:

$$\begin{cases} f_{\omega_{w,3}} = 0\ \text{rpm} & \text{for } t < 10\ s \\ f_{\omega_{w,3}} = a_f = -40\ \text{rpm} & \text{for } t \geq 10\ s \end{cases}$$

fault size $a_f = -40$ rpm

7. **Fault scenario n.7:** sinusoidal fault on the measured flywheel spin rate $\omega_{w,3}$:

$$\begin{cases} f_{\omega_{w,3}} = 0\ \text{rpm} & \text{for } t < 10\ s \\ f_{\omega_{w,3}} = -a_f \sin(2\pi t/T_f) = -0.4\sin(4\pi t)\ \text{rpm} & \text{for } t \geq 10\ s \end{cases}$$

amplitude $a_f = -40$ rpm, period $T_f = 0.5$ sec $(\omega_f = 5\ Hz)$

8. **Fault scenario n.8:** failure of the sensor measuring the flywheel spin rate $\omega_{w,3}$:

$$\begin{cases} f_{\omega_{w,3}} = 0\ \text{rpm} & \text{for } t < 10\ s \\ f_{\omega_{w,3}} = -\omega_{w,3}\ \text{rpm} & \text{for } t \geq 10\ s \end{cases}$$

9. **Fault scenario n.9:** step fault, constant bias on the measured satellite angular velocity ω_1:

$$\begin{cases} f_{\omega_1} = 0 \text{ deg/s} & \text{for } t < 10\ s \\ f_{\omega_1} = a_f = -0.05 \text{ deg/s} & \text{for } t \geq 10\ s \end{cases}$$

fault size $a_f = -0.05$ deg/s

10. **Fault scenario n.10:** lock-in-place fault, stuck measure of the satellite angular velocity ω_1:

$$\begin{cases} \omega_1 = \omega_{1true} \text{ deg/s} & \text{for } t < 10\ s \\ \omega_1 = \omega_1\big|_{t=10s} \text{ deg/s} & \text{for } t \geq 10\ s \end{cases}$$

11. **Fault scenario n.11:** loss of effectiveness of the sensor measuring the angular velocity ω_1:

$$\begin{cases} \omega_1 = \omega_{1true} \text{ deg/s} & \text{for } t < 10\ s \\ \omega_1 = a_f \omega_{1true} = 0.5\omega_{1true} \text{ deg/s} & \text{for } t \geq 10\ s \end{cases}$$

loss of effectiveness (%) $a_f = -50\%$

12. **Fault scenario n.11:** constant quaternion vector $\bar{\mathbf{q}}_f$ multiplying the true quaternion $\bar{\mathbf{q}}_{true}$:

$$\begin{cases} \bar{\mathbf{q}}_f = [0,0,0,1] & \text{for } t < 10\ s \\ \bar{\mathbf{q}}_f = [0.0004363323\ 0\ 0\ 0.9999999048] & \text{for } t \geq 10\ s \end{cases}$$

In this last fault scenario, the equivalent additive fault vector $\bar{\mathbf{F}}_{\bar{\mathbf{q}}_{star}}$ is derived by Eq. (2.55), and hence it depends on the actual quaternion vector in each instant. The fault quaternion $\bar{\mathbf{q}}_f$ corresponds to a rotation of -180 arcsec, or -0.05 deg, around the x_b-axis (roll axis).

These fault scenarios can be divided on the basis of the affected subsystem of the satellite ADCS:

1. Fault scenarios from n.1 to n.8 concern the occurrence of faults affecting the actuators and flywheel spin rate sensors composing the Attitude Control System (ACS) of the satellite;
2. Fault scenarios from n.9 to n.12 concern the occurrence of faults affecting the attitude and angular velocity sensors composing the Attitude Determination System (ADS) of the satellite.

Both abrupt and incipient faults are considered, affecting both satellite actuators and sensors. Moreover, in order to improve the significance and visibility of the considered faults, an attitude change manoeuvre is considered. The considered manoeuvre starts at $t_{man} = 5$ s from the initial attitude quaternion vector $\mathbf{q}_0 = [-0.0570\ \ 0.3180\ \ 0.1663\ \ 0.9316]^T$ to the final attitude quaternion vector $\mathbf{q} = [-0.0367\ \ 0.2975\ \ 0.1774\ \ 0.9374]^T$. If expressed as Euler angles, these vectors represent attitudes equal to $[\phi_0\ \ \theta_0\ \ \psi_0]^T = [-15\ \ 35\ \ 25]^T$ deg and $[\phi\ \ \theta\ \ \psi]^T = [-12\ \ 33\ \ 25]^T$ deg, respectively.

4.4.2 Detection and Isolation of Faults in the Satellite ACS

Let us start considering the occurrence of faults affecting the actuators. In the first fault scenario, *i.e.* a step fault on the control torque $T_{ctrl,2}$, the occurred fault can be detected and isolated by exploiting the residual cross-checking procedure illustrated in Section 4.3. Figs. 4.1 and 4.2 show the disturbance decoupled diagnostic signals provided by the banks RB1 and RB2 of local and global residual filters, respectively. For all the fault scenarios related to the ACS, only the residuals of RB1 and RB2 are shown, since the fault detection and isolation can be realized by means of these two banks of residual filters. The residual filters of RB3 and RB4 are, by construction, not sensitive to any fault affecting the actuators or sensors of the ACS since they are based on the kinematic equations of the spacecraft model. The selected thresholds are depicted in red in the figures. After the fault occurrence, the local residual r_2 sensitive to the faults affecting the second actuator subsystem show a deviation from the initial fault-free condition and it exceeds the selected threshold. On the contrary, the other local residuals r_1, r_3 and r_4 do not exceed their thresholds, meaning that the first, third and fourth actuator subsystems are properly working. On the other hand, it can be seen that the global residuals r_5, r_6, r_7, r_8 and r_9, which are not sensitive to any fault affecting the attitude control inputs maintain values near to zero and do not exceed the selected thresholds. Hence, the threshold exceeding of only this specific local residual allows the isolation of the faulty actuator subsystem, whilst all the global residuals not exceeding their thresholds mean that the occurred fault in the second actuator subsystem is actually affecting the actuated control torque.

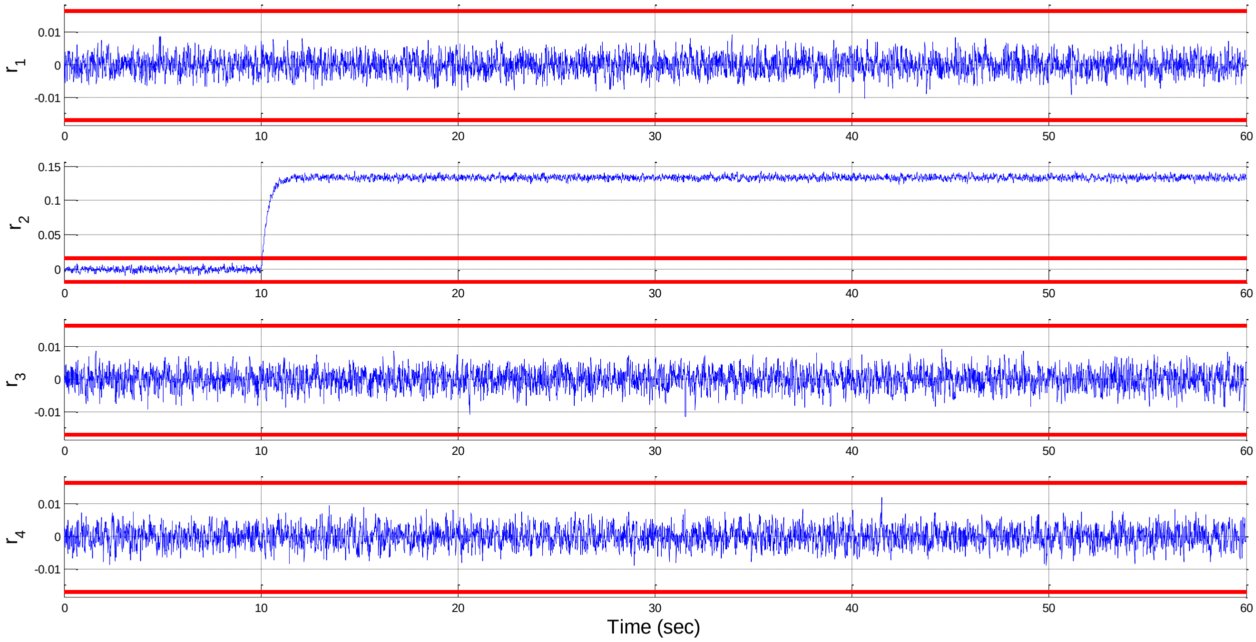


Figure 4.1 - Fault scenario n.1: local residuals of RB1 sensitive to fault inputs associated to physical actuator and flywheel spin rate sensor faults in case of step fault on the control input $T_{ctrl,2}$.

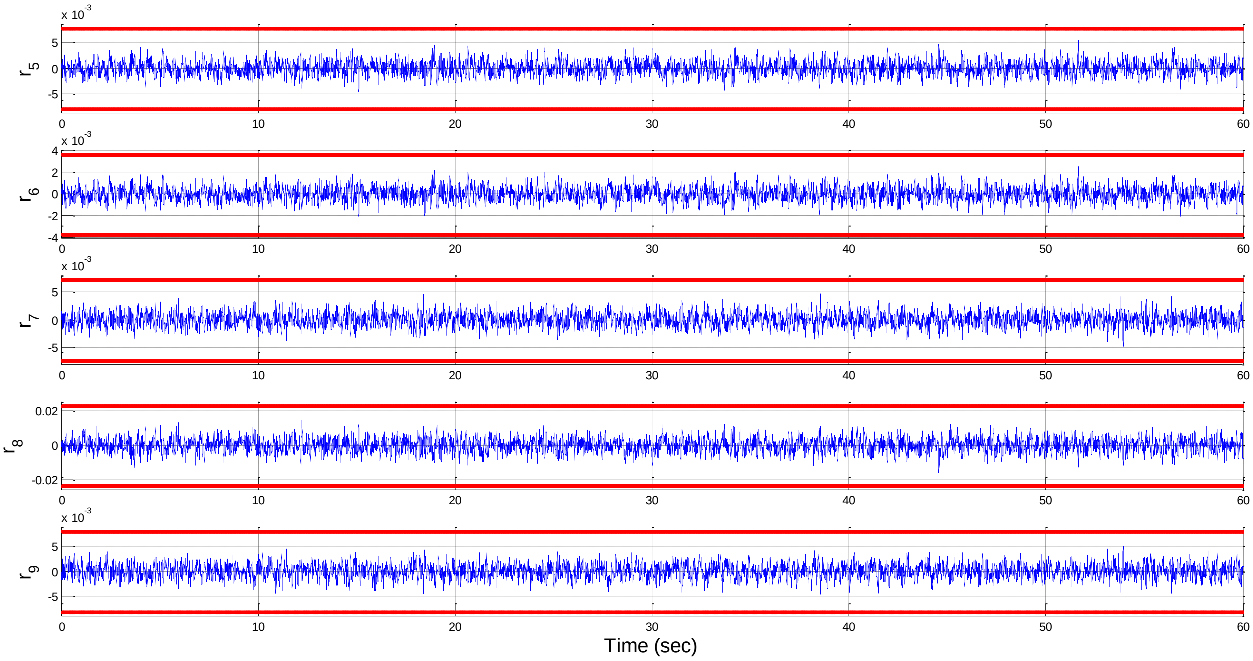


Figure 4.2 - Fault scenario n.1: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of step fault on the control input $T_{ctrl,2}$.

A similar behaviour for both the global and local residual signals is shown in Figs. 4.3, 4.4, 4.5 and 4.6 in case of the second and third fault scenarios, *i.e.* the sinusoidal and rectangular pulse faults affecting the attitude control input $T_{ctrl,2}$. Therefore, in both these scenarios the occurred fault can be detected and isolated by means of the described cross-checking procedure.

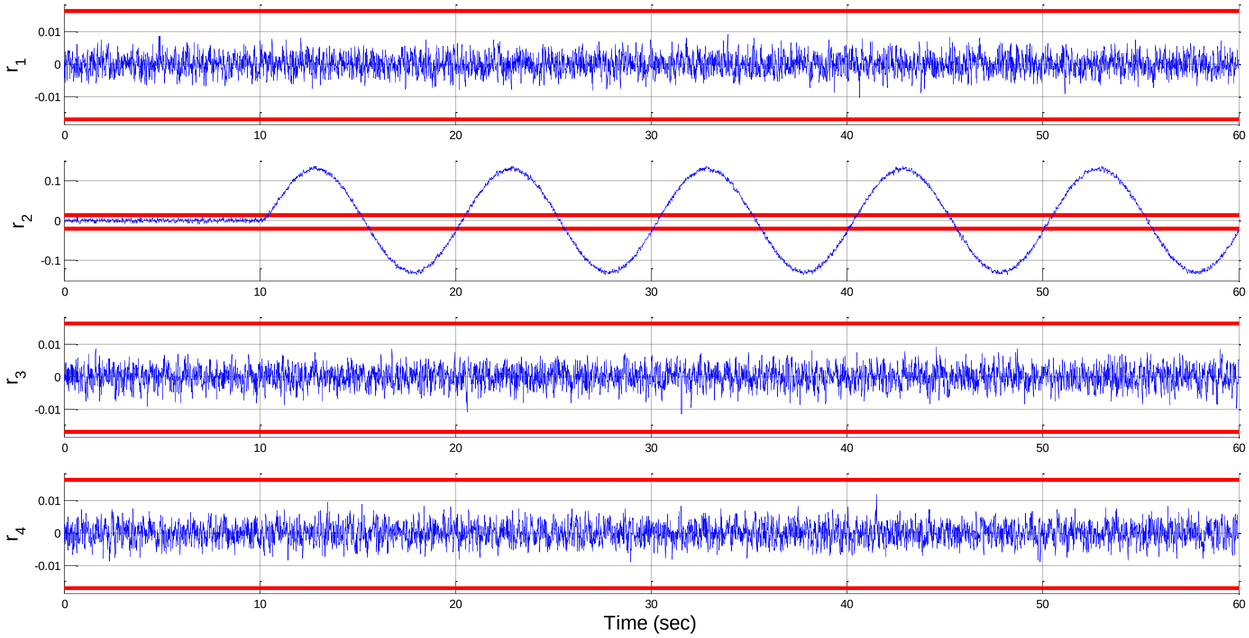


Figure 4.3 - Fault scenario n.2: local residuals of RB1 sensitive to fault inputs associated to physical actuator and flywheel spin rate sensor faults in case of sinusoidal fault on the control input $T_{ctrl,2}$.

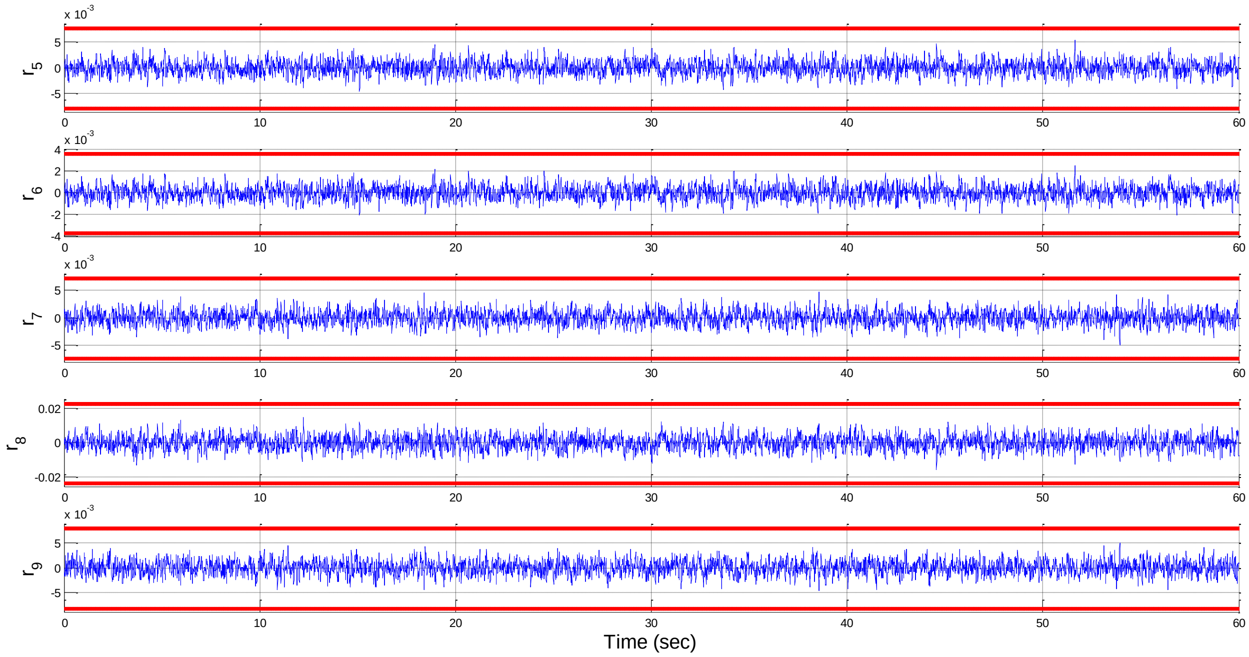


Figure 4.4 - Fault scenario n.2: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of sinusoidal fault on the control input $T_{ctrl,2}$.

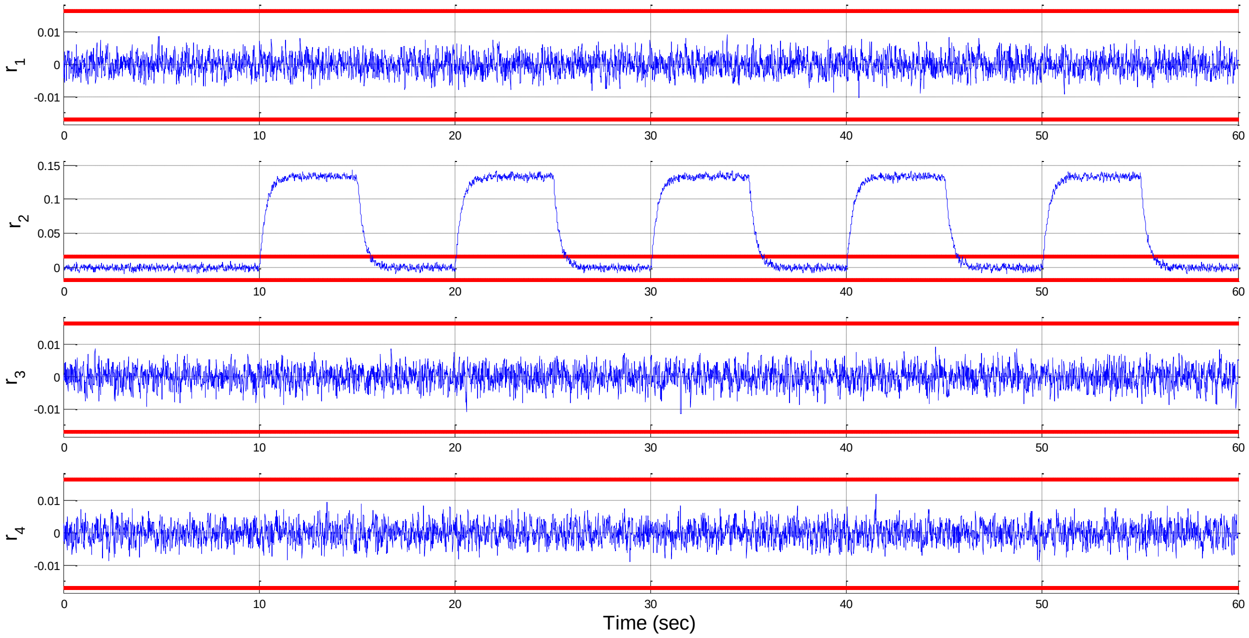


Figure 4.5 - Fault scenario n.3: local residuals of RB1 sensitive to fault inputs associated to physical actuator and flywheel spin rate sensor faults in case of rectangular pulse fault on the control input $T_{ctrl,2}$.

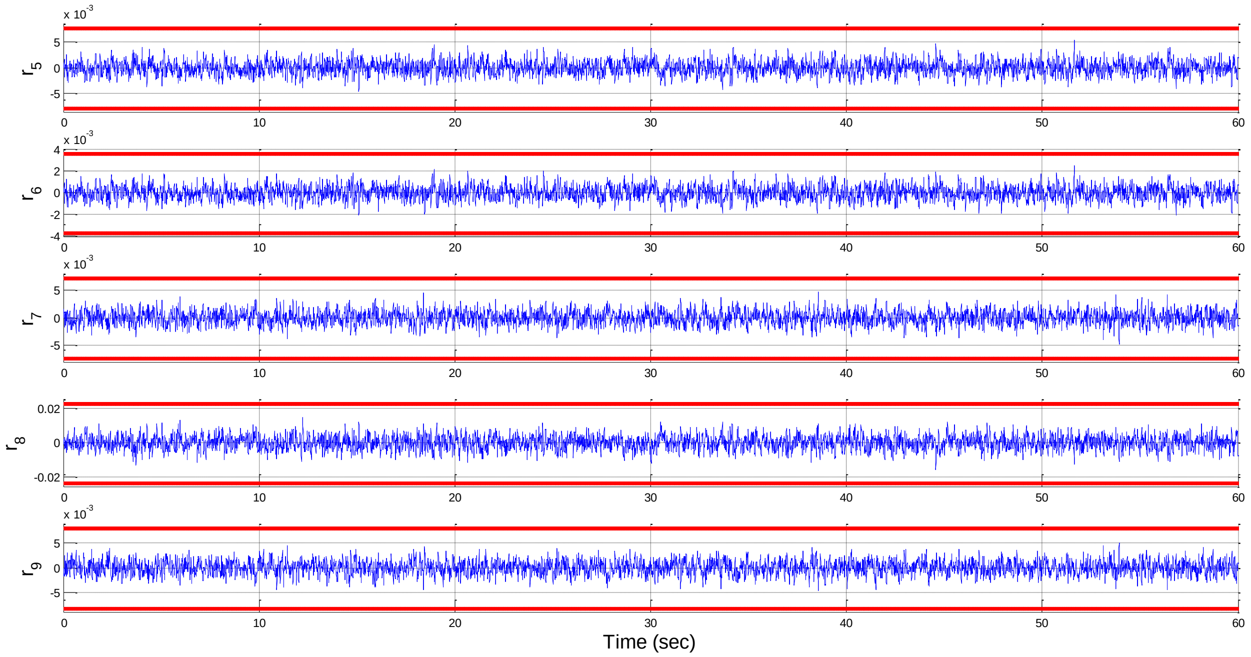


Figure 4.6 - Fault scenario n.3: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of rectangular pulse fault on the control input $T_{ctrl,2}$.

In these three considered fault scenarios, a prompt and accurate detection and isolation of the abrupt fault is realized, with detection times abundantly smaller than $t_{\det} = 1$ s. On the other hand, in Figs. 4.7 and 4.8, the occurrence of an incipient ramp fault is considered. Again, only the second local residual of RB1 presents a deviation from its fault-free condition, and it exceeds the selected threshold. However, the detection time is significantly larder due to the slow increasing of the occurred ramp fault, and thus of the sensitive residual. The fault is effectively detected at about $t_{\det} = 30$ s.

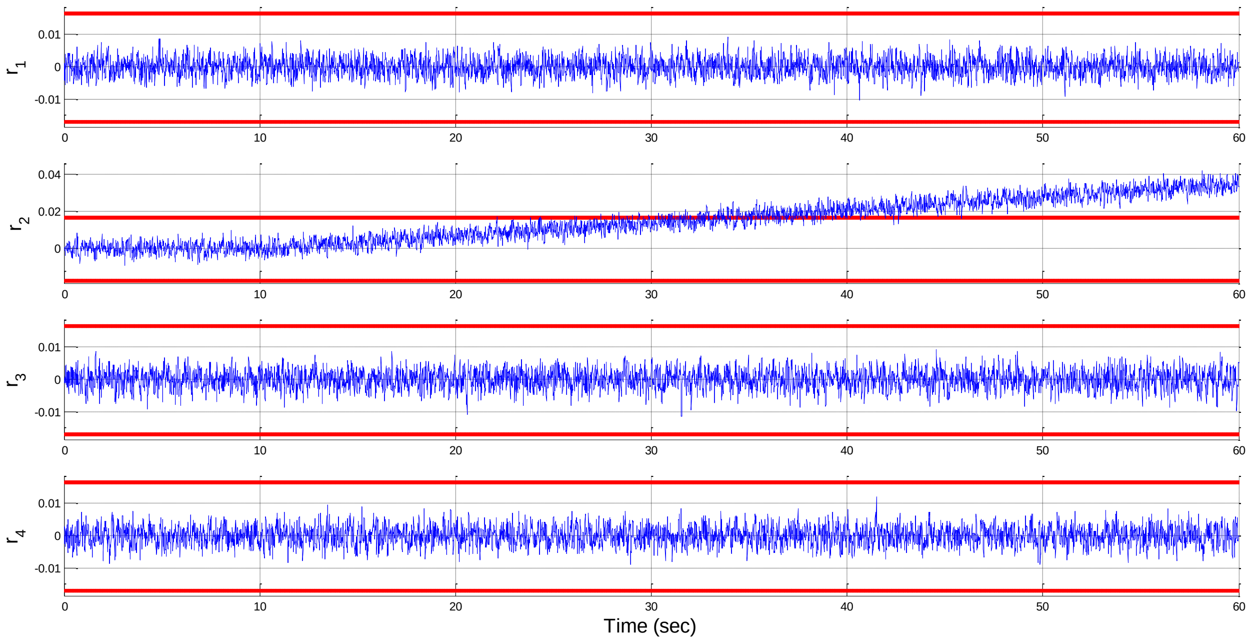


Figure 4.7 - Fault scenario n.4: local residuals of RB1 sensitive to fault inputs associated to physical actuator and flywheel spin rate sensor faults in case of ramp fault on the control input $T_{ctrl,2}$.

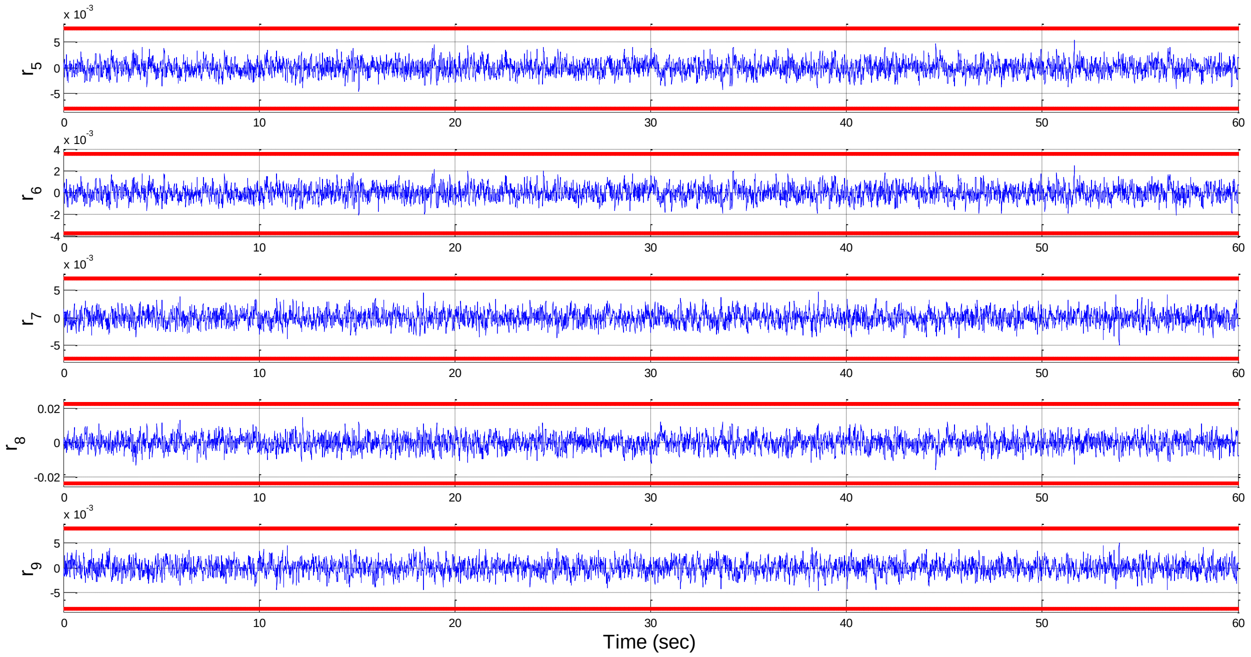


Figure 4.8 - Fault scenario n.4: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of ramp fault on the control input $T_{ctrl,2}$.

Finally, the total failure of the actuator providing the attitude control input $T_{ctrl,2}$ is considered. Figs. 4.9 and 4.10 show the behaviour of the local and global residuals of RB1 and RB2 in case of failure of the second actuator.

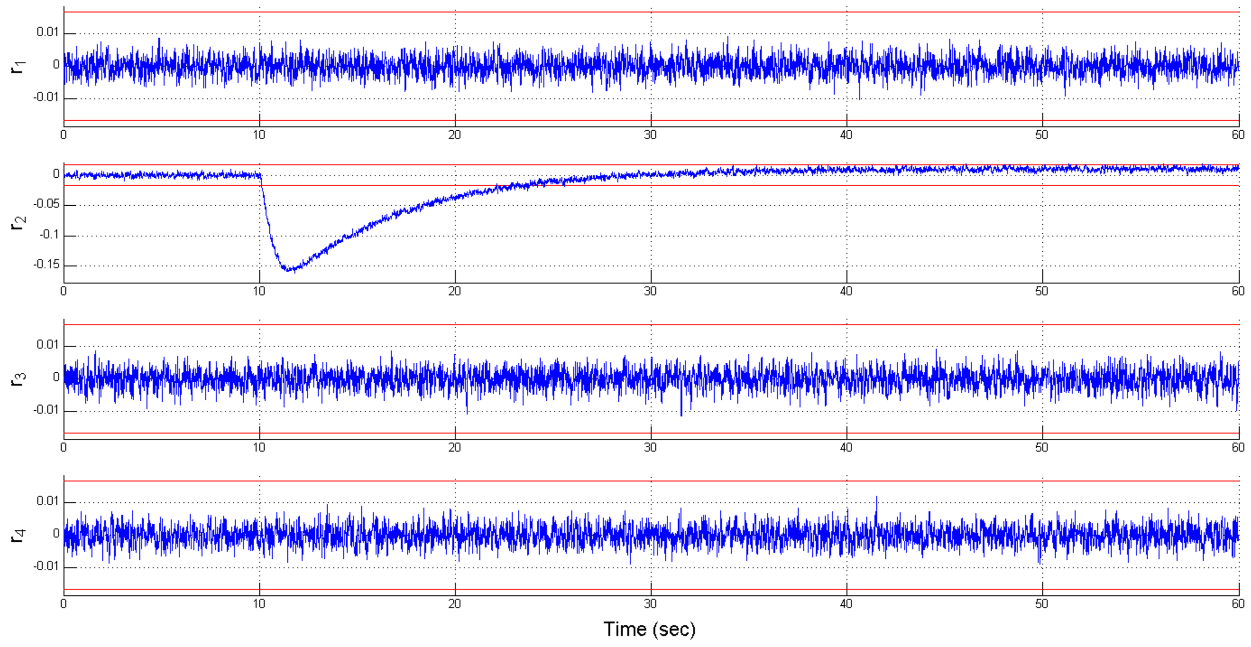


Figure 4.9 - Fault scenario n.5: local residuals of RB1 sensitive to fault inputs associated to physical actuator faults and corresponding flywheel spin rate sensor faults in case of failure of the second actuator.

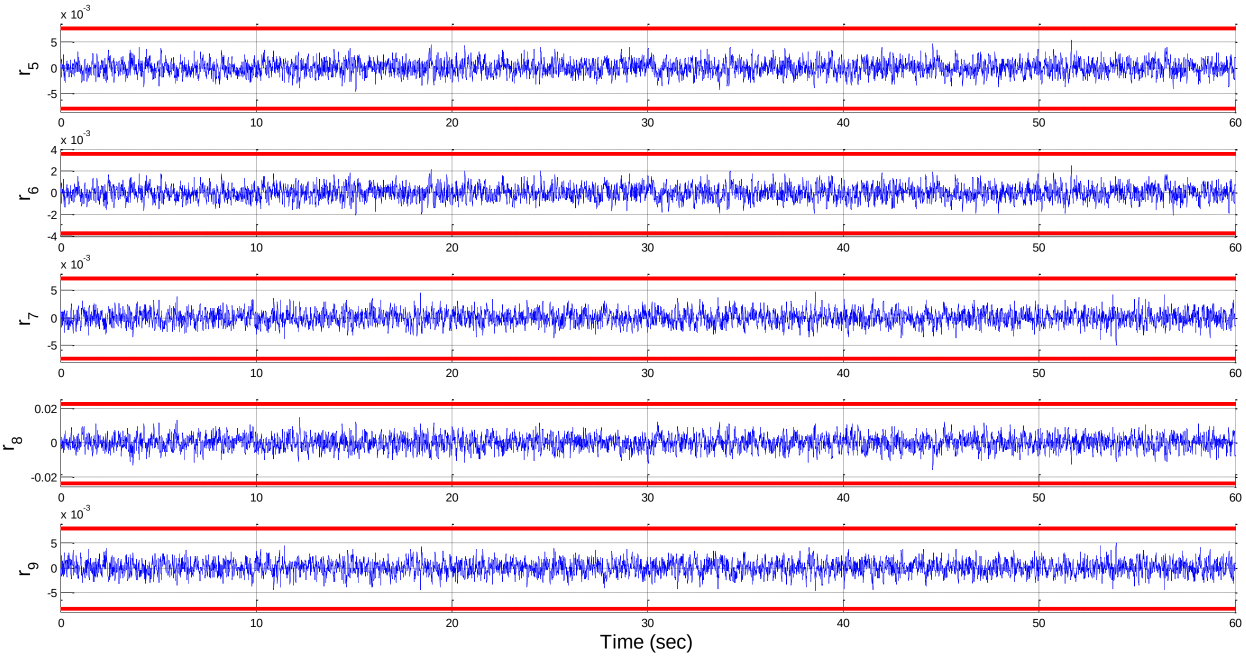


Figure 4.10 - Fault scenario n.5: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of failure of the second actuator.

Considering the occurrence of faults affecting the sensor measurement of $\omega_{w,3}$, the occurred fault can be detected and isolated again by exploiting the residual cross-checking procedure illustrated in Section 4.3. Figs. 4.11 and 4.12 show the diagnostic signals provided by RB1 and RB2 respectively, in case of the sixth fault scenario, *i.e.* a step fault affecting the measurement of $\omega_{w,3}$. Similarly to the case of actuator fault, after the occurrence of a fault in the third flywheel spin rate sensor, the local residual r_3 sensitive to the faults affecting the third actuator subsystem shows a deviation from the initial fault-free condition and it exceeds the selected thresholds. On the contrary, the other local residuals r_1, r_2 and r_4 do not exceed their thresholds, meaning that the first, second and fourth actuator subsystems are properly working. On the other hand, now it can be seen that now the global residuals r_5, r_6, r_7, r_8 and r_9, which are in general not sensitive only to faults affecting the control inputs, now exceed the selected thresholds. Hence, the threshold exceeding of this specific local residual allows the isolation of the faulty actuator subsystem, whereas the threshold exceeding of the global residuals means that the occurred fault in the third actuator subsystem is actually affecting the respective flywheel spin rate sensor. It is worth noting that the residuals result to be sensitive mainly to the mathematical fault input $f_{x_3,3} = \dot{\omega}_{w,3_{measured}} - \dot{\omega}_{w,3_{true}}$ associated with the time derivative of the occurred flywheel spin rate sensor fault. Hence, in case of constant fault (*i.e.* a step fault), the stabilizing residual feedback in the residual filter models cause the diagnostic signals to return to zero in a certain time after the fault occurrence, with a convergence speed depending on the value of the selected filter gains $K > 0$.

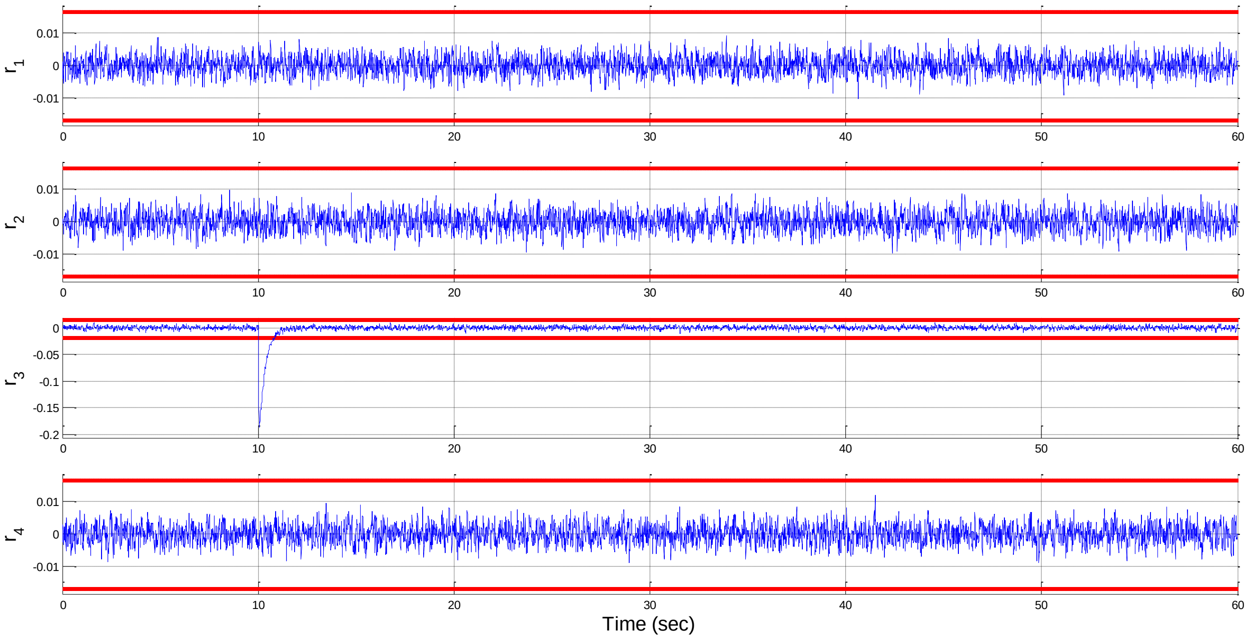


Figure 4.11 - Fault scenario n.6: local residuals of RB1 sensitive to fault inputs associated to physical actuator faults and corresponding flywheel spin rate sensor faults in case of step fault on the sensor output $\omega_{w,3_{measured}}$.

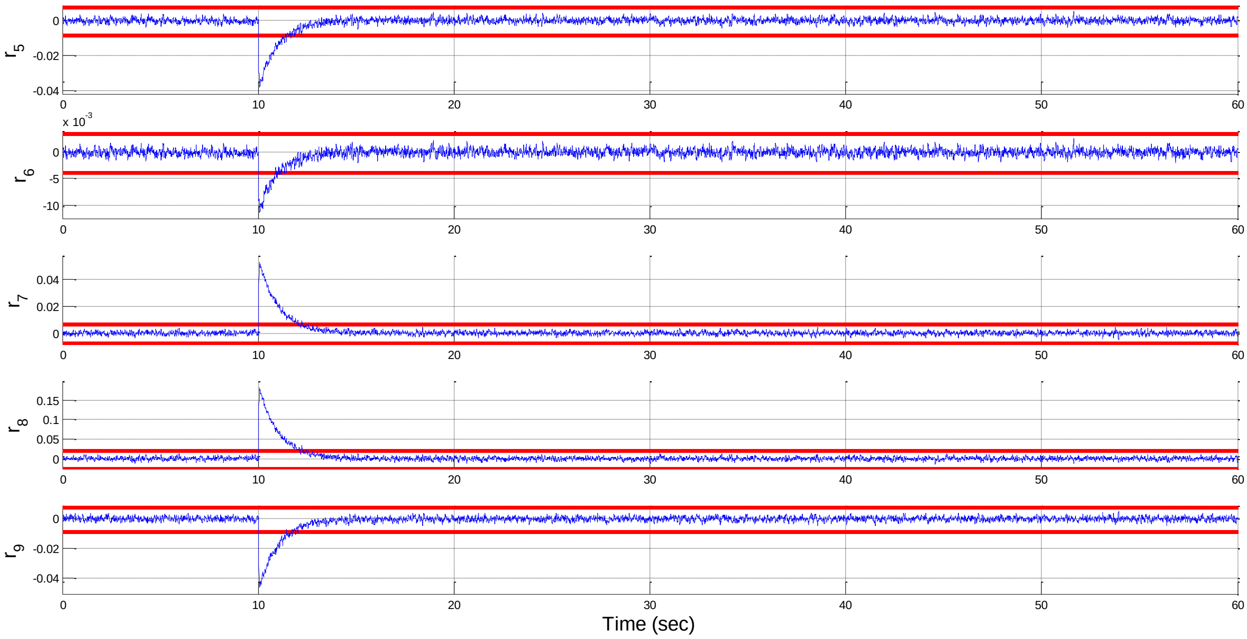


Figure 4.12 - Fault scenario n.6: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of step fault on the sensor output $\omega_{w,3_{measured}}$.

Figs. 4.13, 4.14, 4.15 and 4.16 show the local and global residuals in case of the seventh and eighth fault scenarios, *i.e.* a sinusoidal fault and the total failure of the sensor providing the measurement of $\omega_{w,3}$, respectively.

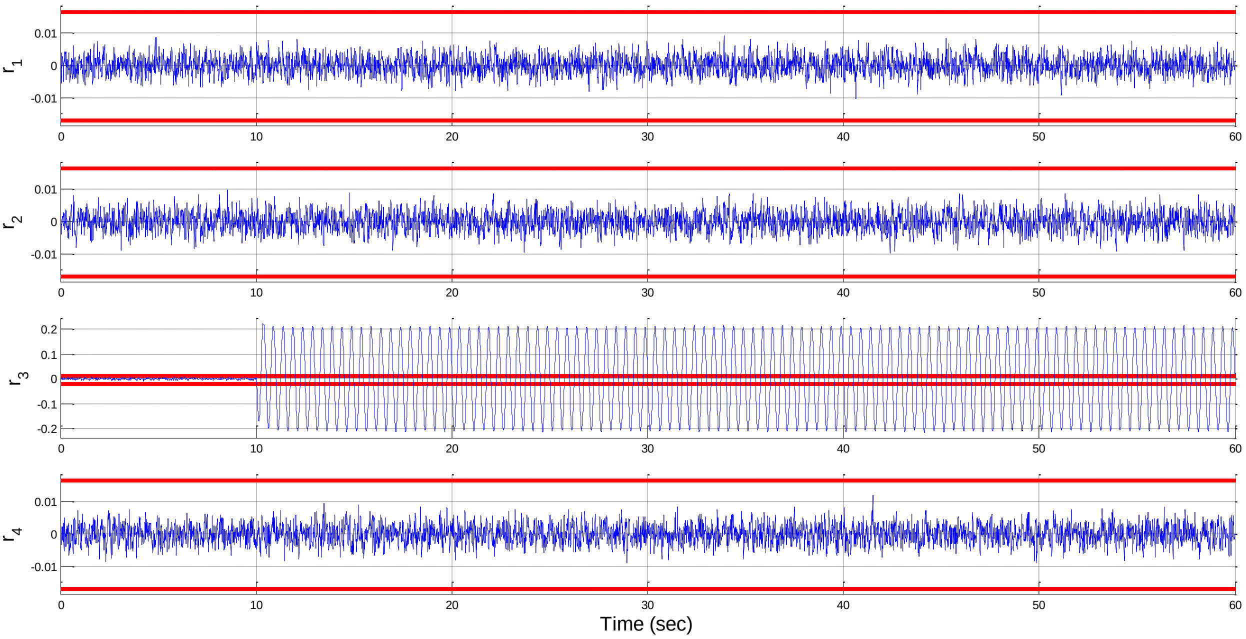


Figure 4.13 - Fault scenario n.7: local residuals of RB1 sensitive to fault inputs associated to physical actuator faults and corresponding flywheel spin rate sensor faults in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$.

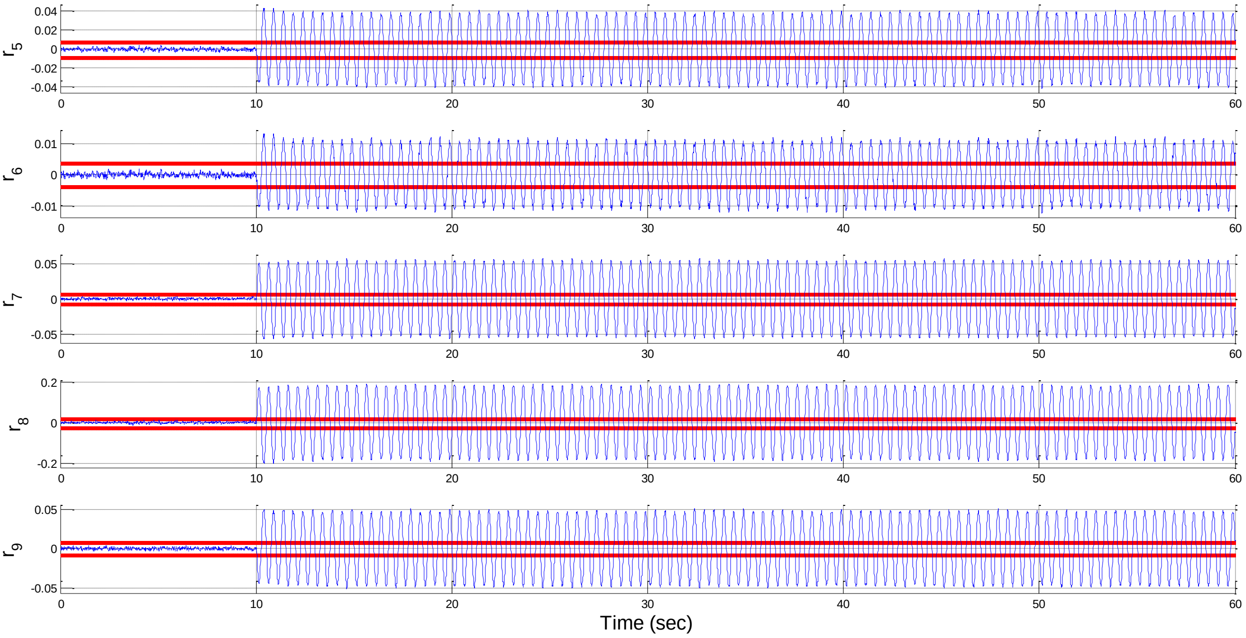


Figure 4.14 - Fault scenario n.7: global residuals of RB2 not sensitive to any fault input associated to physical actuator faults in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$.

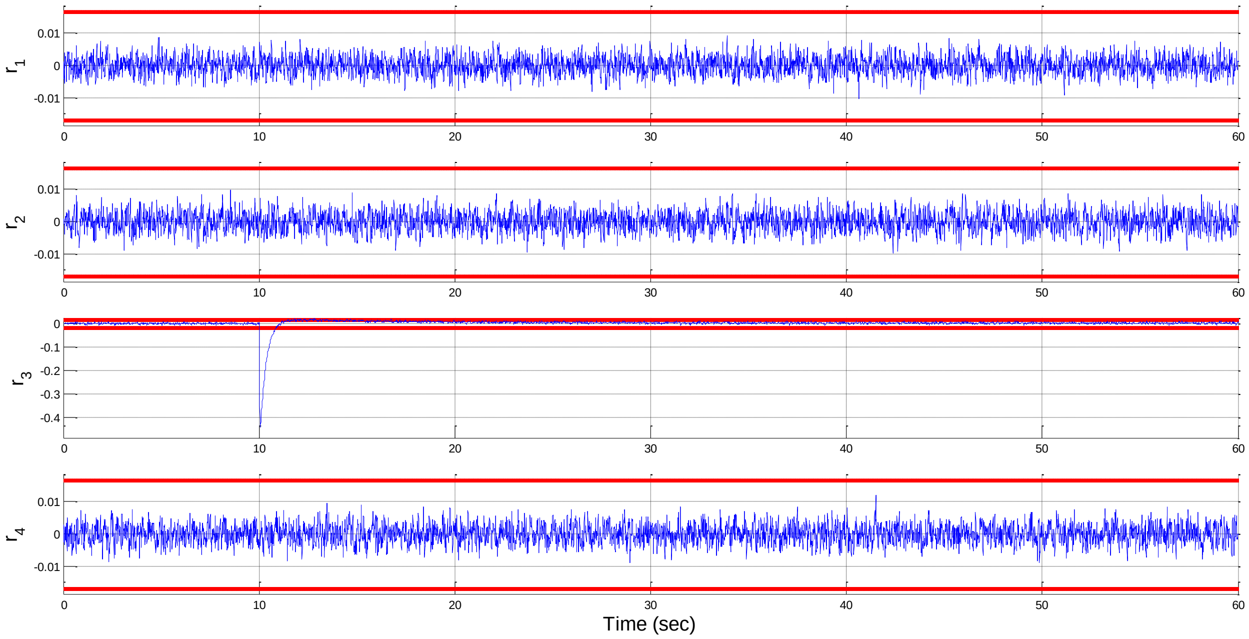


Figure 4.15 - Fault scenario n.8: local residuals of RB1 sensitive to fault inputs associated to physical actuator faults and corresponding flywheel spin rate sensor faults in case of failure of the sensor measuring $\omega_{w,3}$.

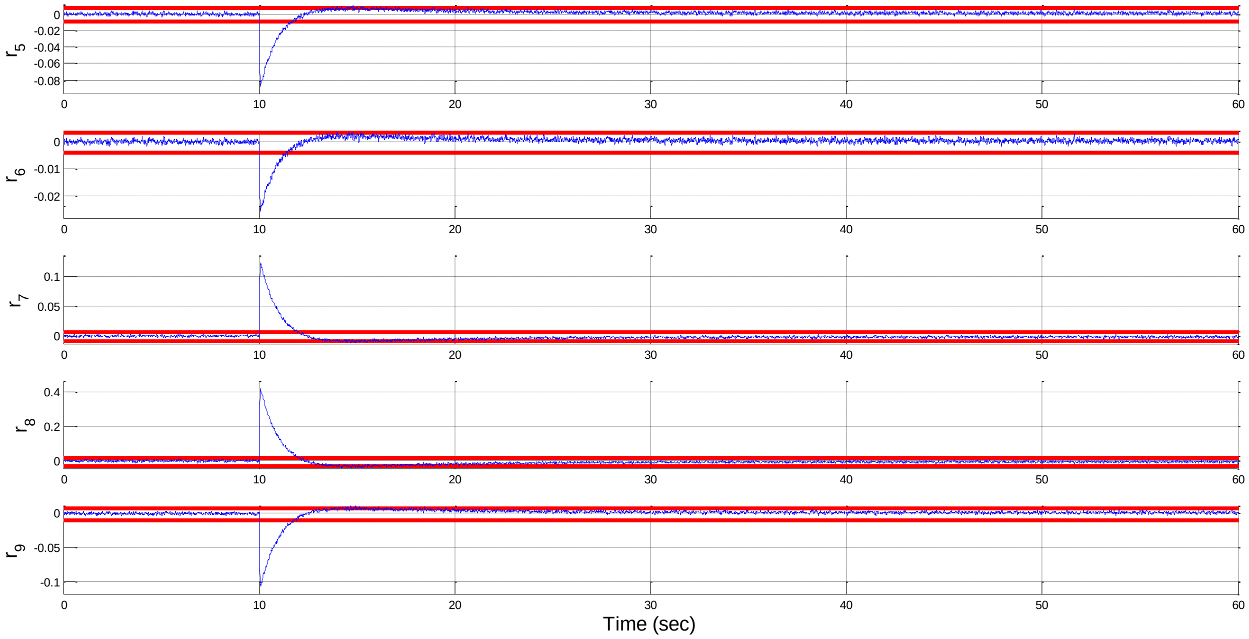


Figure 4.16 - Fault scenario n.8: global residuals of RB2 not sensitive to any fault input associated to physical actuator torque faults. In this case, all the global residuals exceed the selected thresholds in case of failure of the sensor measuring $\omega_{w,3}$.

4.4.3 Detection and Isolation of Faults in the Satellite ADS

Now, considering the occurrence of faults affecting the sensors of the satellite ADS, the ninth fault scenario, *i.e.* a step fault on the sensor measurement of $\omega_{x,i}$, the occurred fault can be detected and isolated again by exploiting the residual cross-checking procedure illustrated in Section 4.3.

In this case, only the residuals provided by RB3 and RB4 are effectively exploited for the detection and isolation of faults affecting the sensors of the ADS.

Figs. 4.17, 4.18 and 4.19 show the diagnostic signals provided by the residual filters of RB3. Each of the nine obtained residual signals results to be sensitive to faults affecting a specific couple of angular velocity sensors. Hence, organizing these signals in a generalized scheme, in each of these three figures the residual signals are grouped depending on the specific couple of angular velocity sensor faults they are sensitive to. The signals in Fig. 4.17 are sensitive to possible faults affecting the measurements of $\omega_{x,i}$ and $\omega_{y,i}$, the signals in Fig. 4.18 are sensitive to faults affecting the measurements of $\omega_{x,i}$ and $\omega_{z,i}$, and finally, the signals of Fig. 4.19 are sensitive to faults affecting the measurements of $\omega_{y,i}$ and $\omega_{z,i}$. As it can be seen by exploiting the cross-check of these three subsets of signals, all the residual signals of Fig. 4.19 does not exceed their thresholds, thus meaning that the occurred fault is not affecting the measurements of $\omega_{y,i}$ and $\omega_{z,i}$, while the residuals in both Figs. 4.17 and 4.18 exceed their thresholds. As already stated, it can be proven that at least a residual signal of three exceeds its threshold in any attitude condition of the spacecraft if the actual sensor fault is not zero for each subset of three sensitive residuals. Hence, the detection and isolation of the fault occurred on the sensor measuring $\omega_{x,i}$ can be easily achieved.

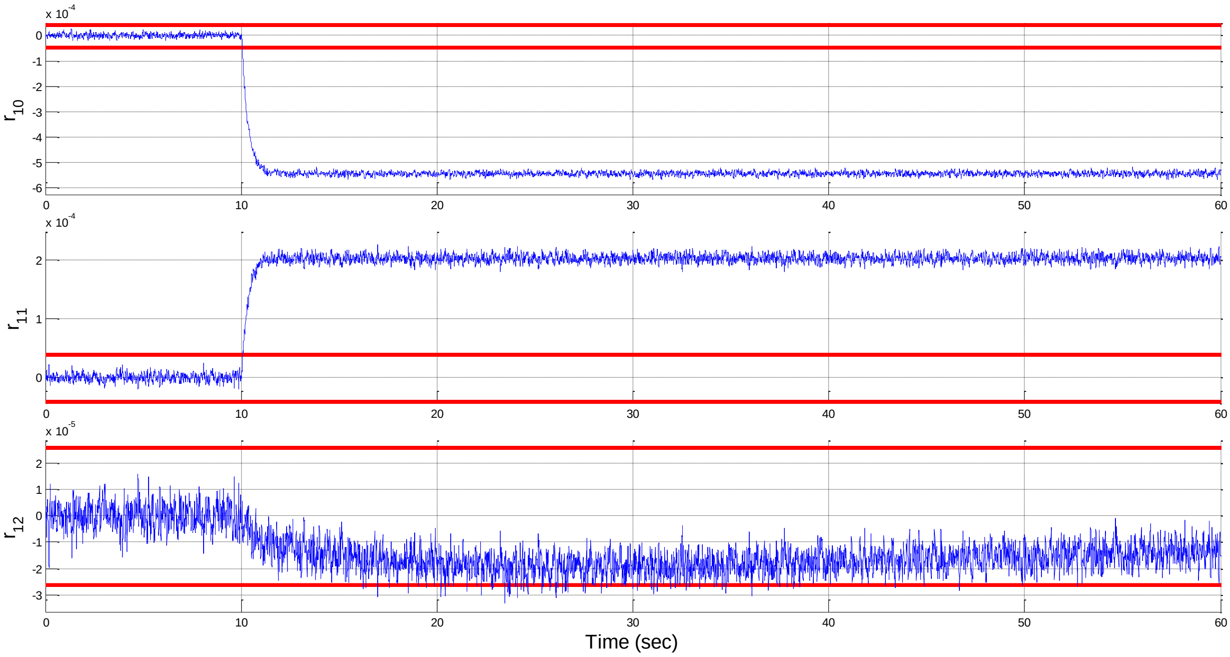


Figure 4.17 - Fault scenario n.9: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

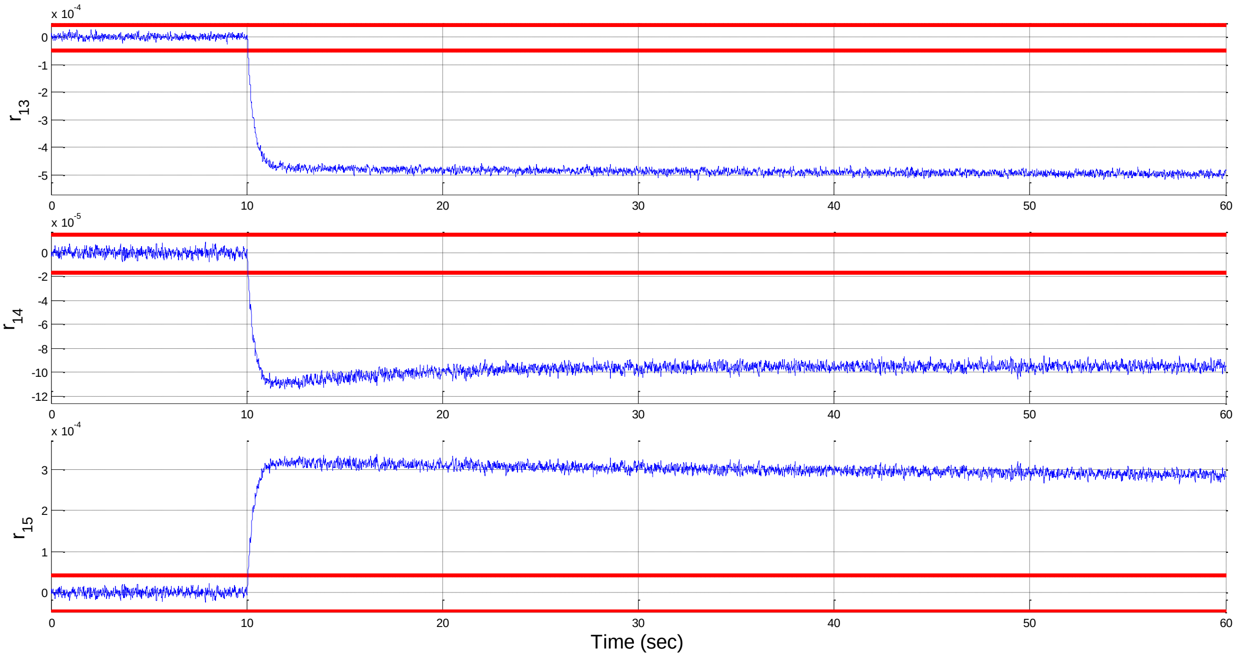


Figure 4.18 - Fault scenario n.9: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

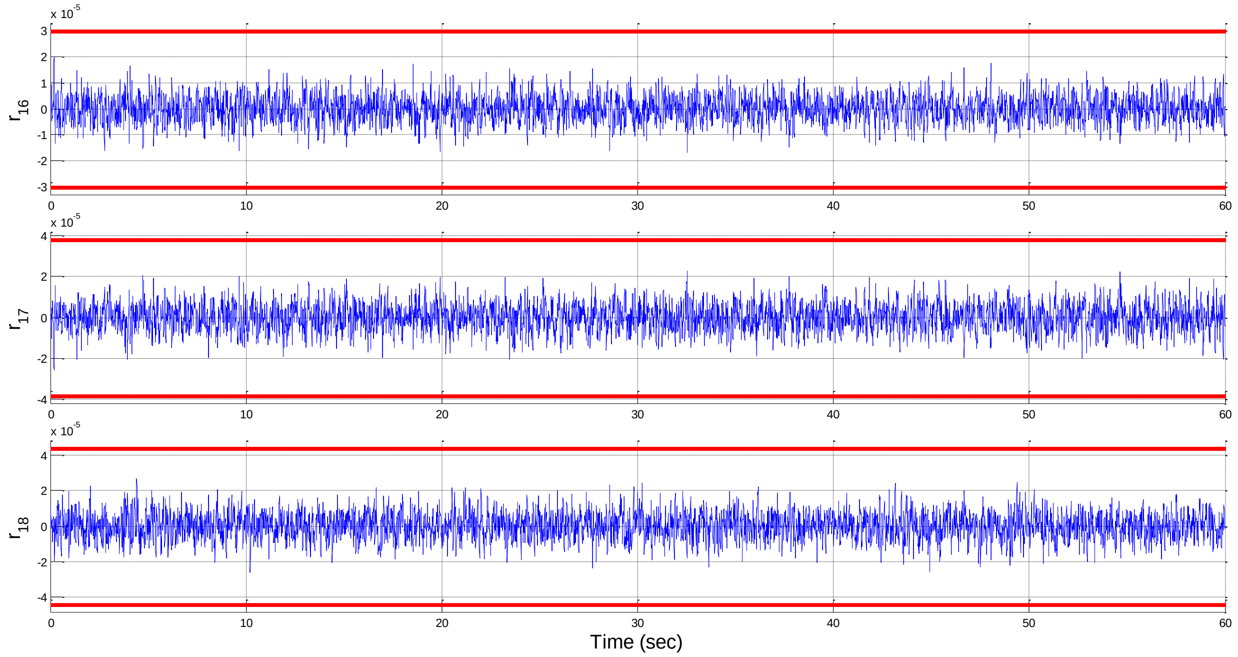


Figure 4.19 - Fault scenario n.9: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{y,i}$ and $\omega_{z,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

Figs. 4.20, 4.21 and 4.22 show the same sets of residual signals but provided by the residual filters of RB4. Since the residual filters of these two banks are based on the same models and exploits the same set of angular velocity measurements, with the assumption of single fault, the combined cross-check of the residuals of the two banks gives a confirmation of the correct fault isolation. In fact, in both RB3 and RB4 the corresponding signals have the same behaviours and each bank of diagnostic signals indicate the occurrence of a fault affecting the sensor measuring $\omega_{y,i}$.

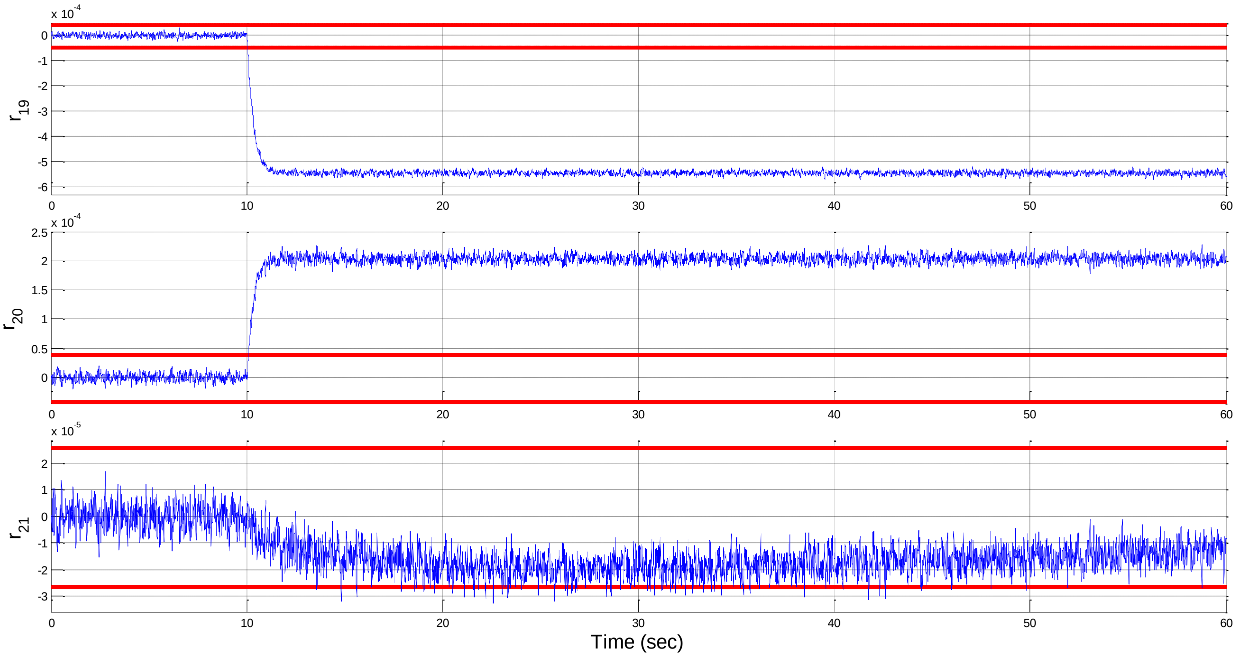


Figure 4.20 - Fault scenario n.9: residuals of RB4, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

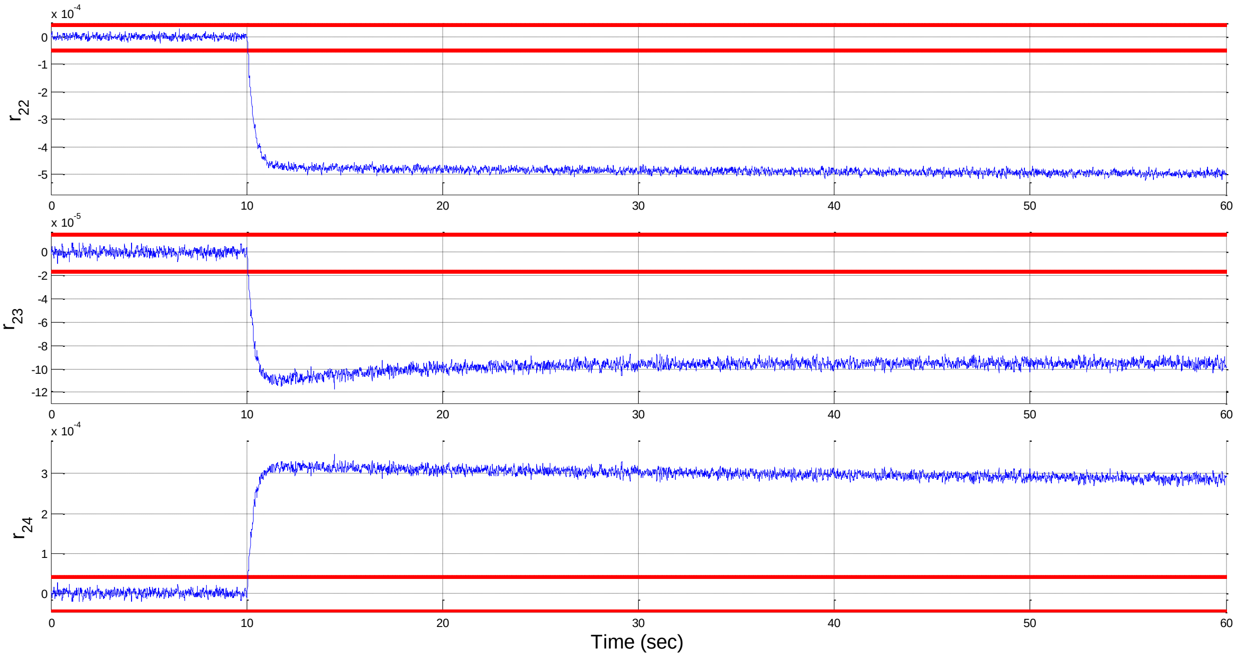


Figure 4.21 - Fault scenario n.9: residuals of RB4, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

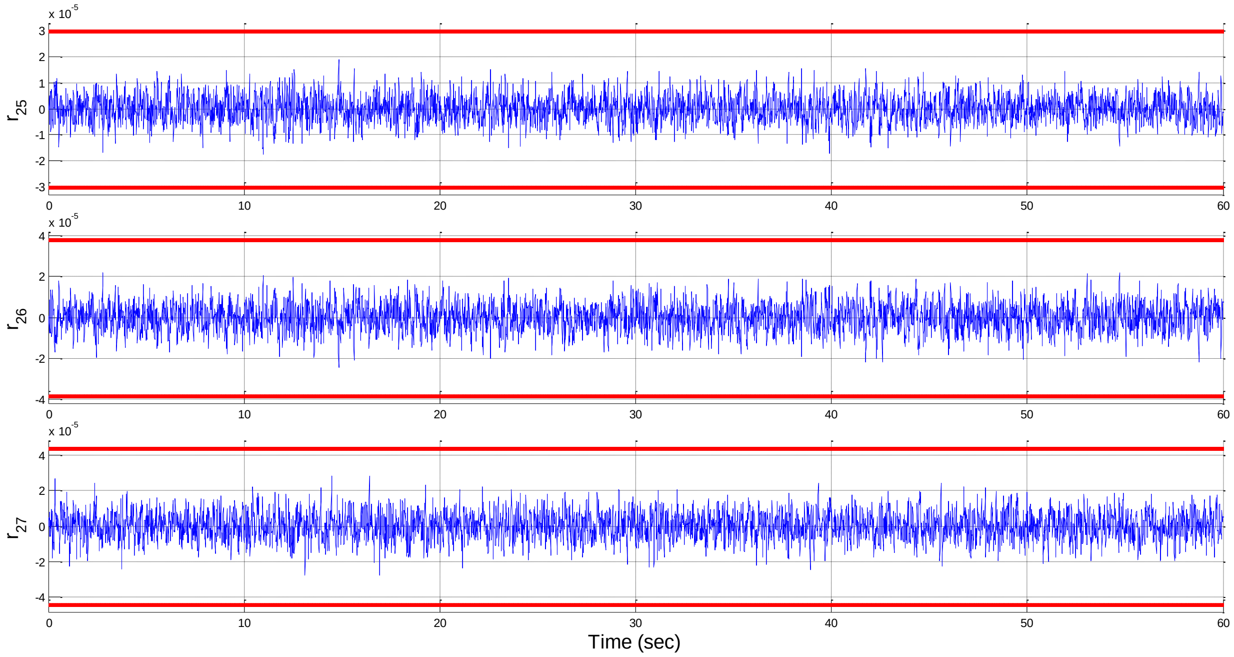


Figure 4.22 - Fault scenario n.9: residuals of RB4, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{y,i}$ and $\omega_{z,i}$, in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

Figs. 4.23, 4.24, 4.25, 4.26, 4.27 and 4.28 show the detection and isolation results provided by RB3 in case of the tenth and eleventh fault scenarios, *i.e.* the lock-in-place (stuck) and loss-of-effectiveness faults on the sensor measuring $\omega_{x,i}$, respectively. The results provided by RB4 are similar, and hence omitted for simplicity.

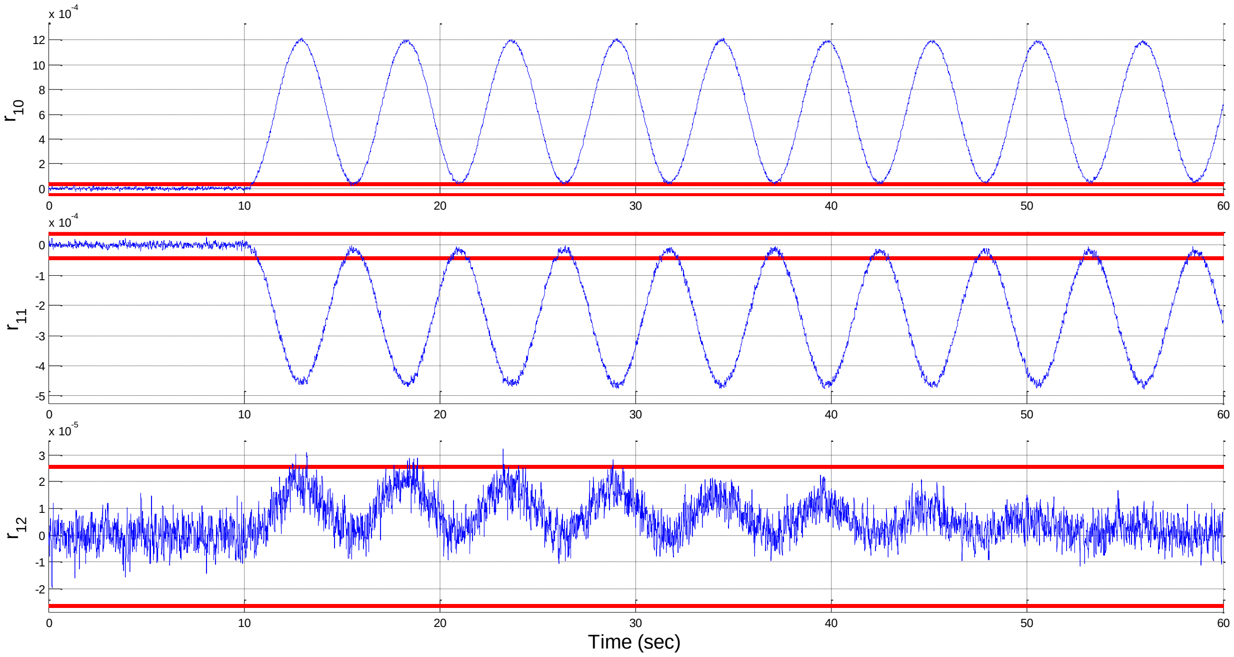


Figure 4.23 - Fault scenario n.10: residuals of RB4, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$, in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

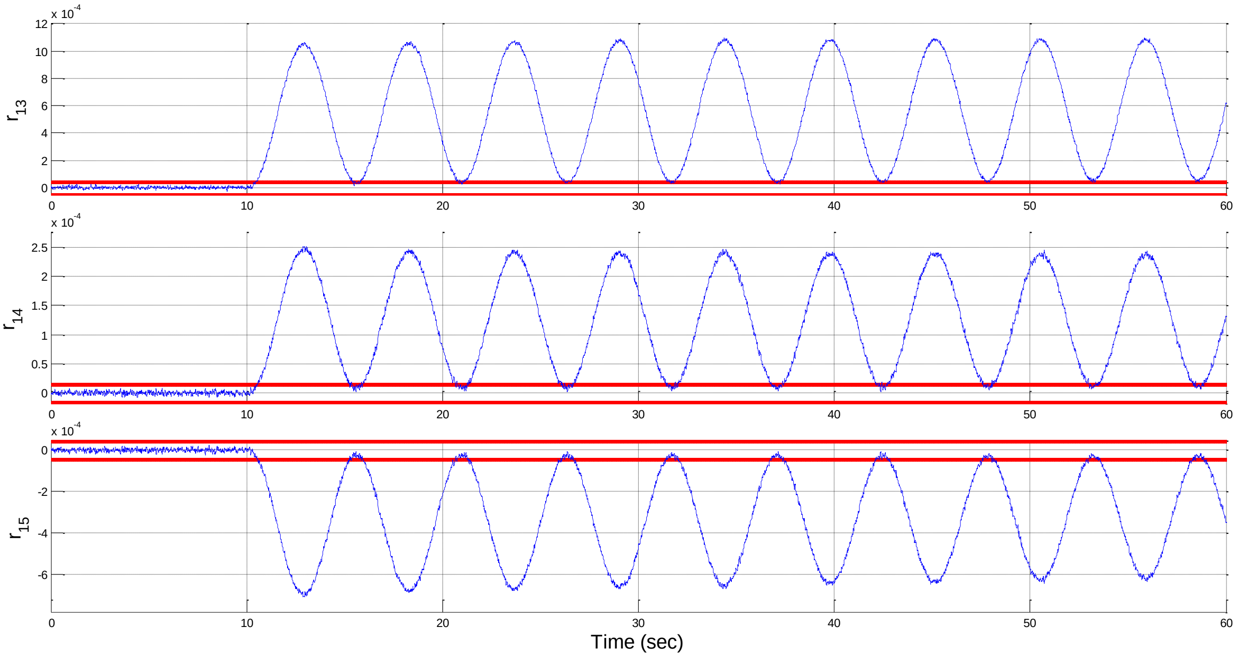


Figure 4.24 - Fault scenario n.10: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$, in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

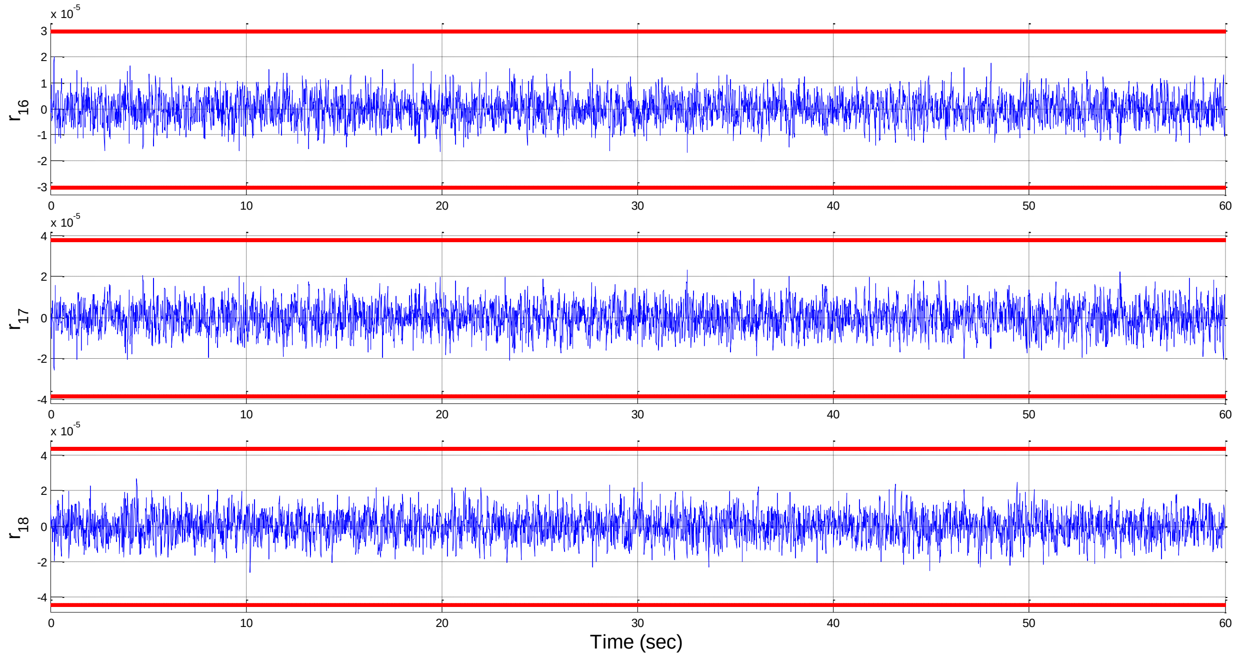


Figure 4.25 - Fault scenario n.10: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{y,i}$ and $\omega_{z,i}$, in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

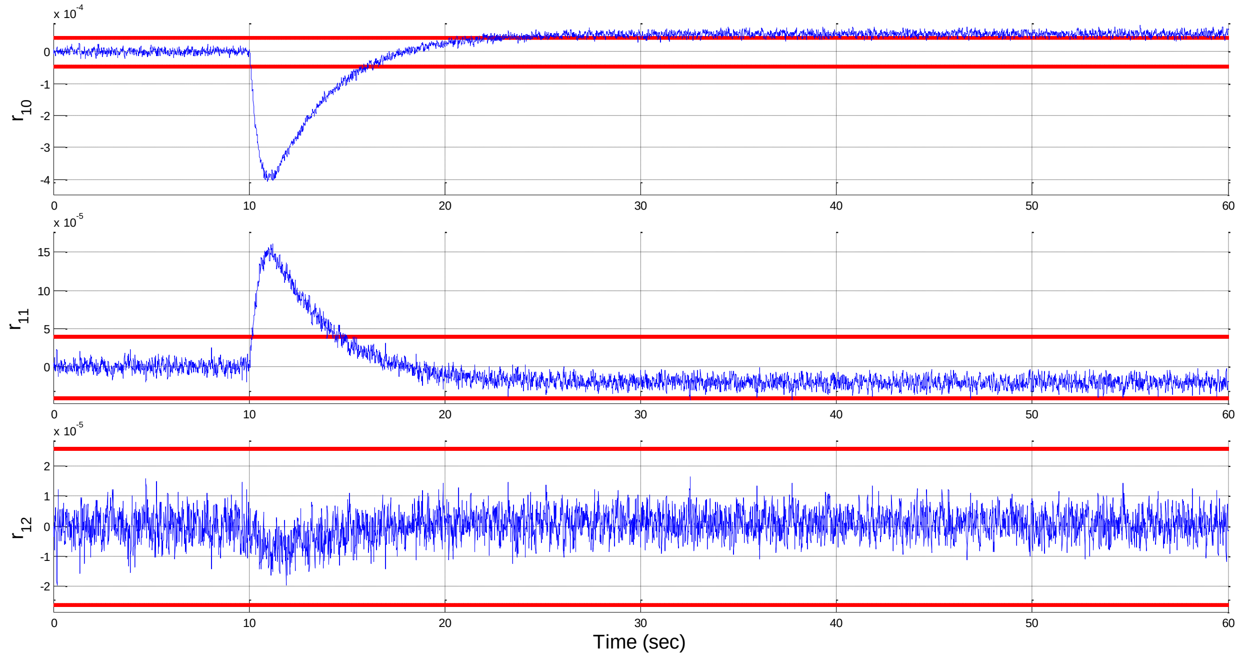


Figure 4.26 - Fault scenario n.11: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$, in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

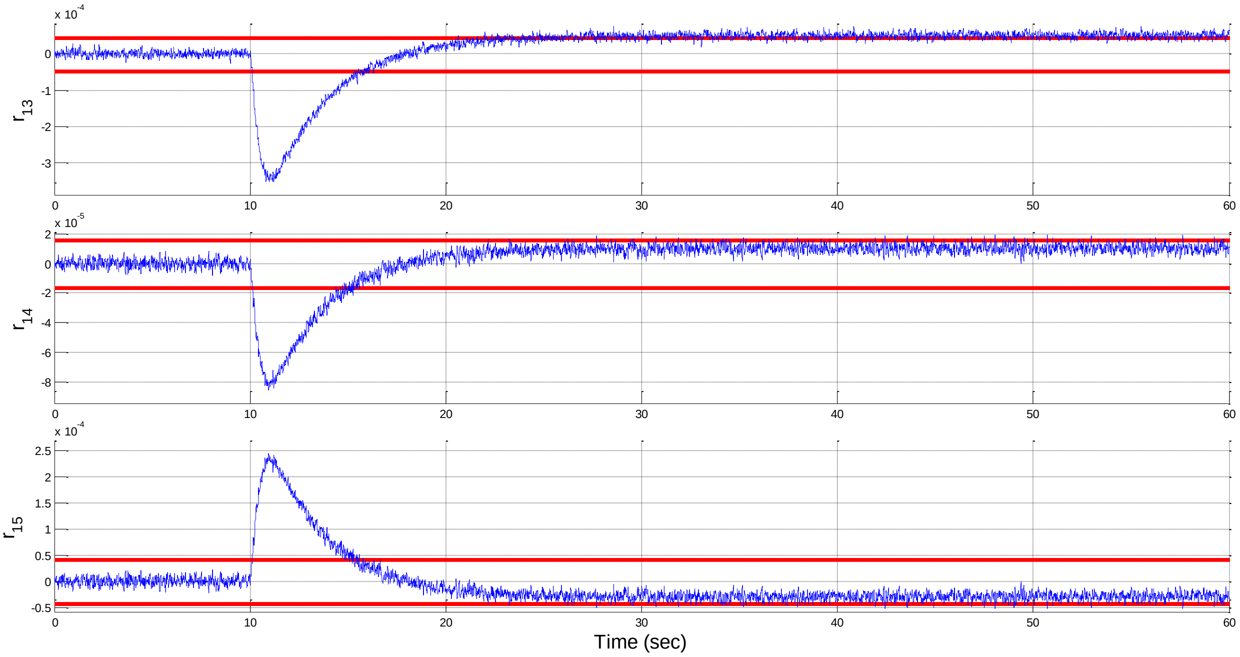


Figure 4.27 - Fault scenario n.11: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$, in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

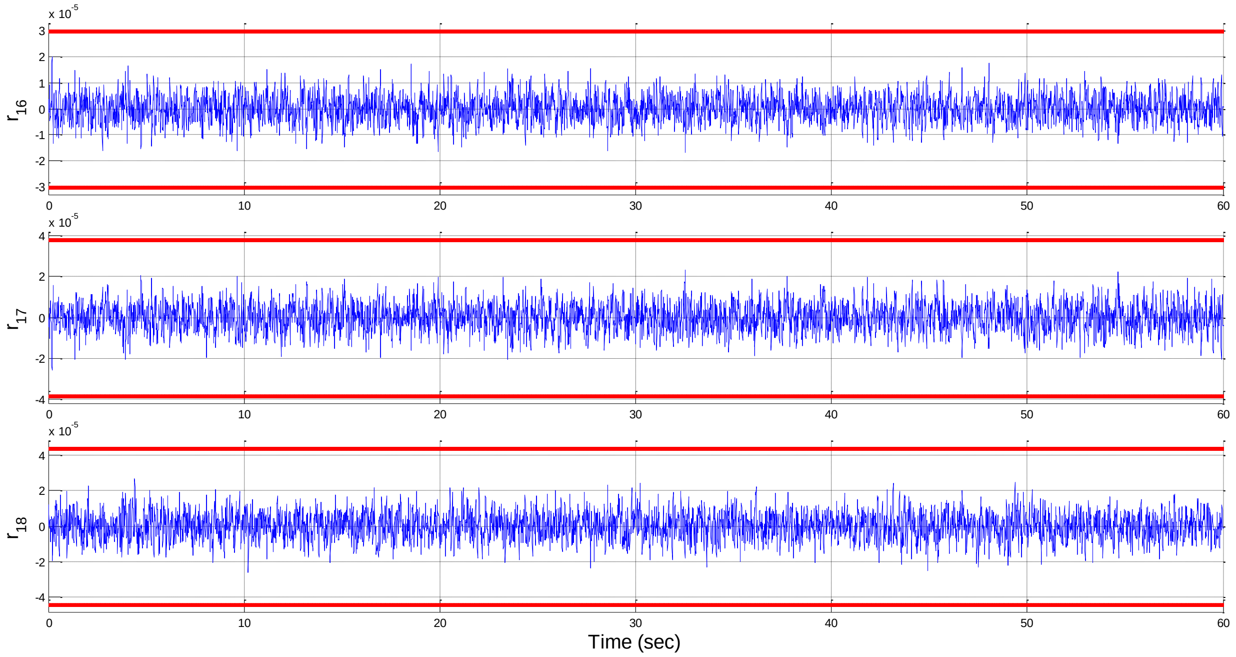


Figure 4.28 - Fault scenario n.11: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{y,i}$ and $\omega_{z,i}$, in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

Finally, considering the twelfth fault scenario, *i.e.* the additive fault $\overline{\mathbf{F}}_{\overline{\mathbf{q}}_{star1}}$ on the sensor measurement of $\overline{\mathbf{q}}_{star1}$, the occurred fault can be detected and isolated again by exploiting the residual cross-checking of the sets of nine diagnostic signals provided by the residual filters of the two banks RB3 and RB4. As it can be seen, Figs. 4.29, 4.30 and 4.31 show the diagnostic signals provided by RB3. These signals results to be sensitive to faults affecting the first attitude sensor. Hence, they exceed their thresholds because the nine residual filters of RB3 are sensitive to faults affecting the attitude measurements provided by the first attitude sensor only.

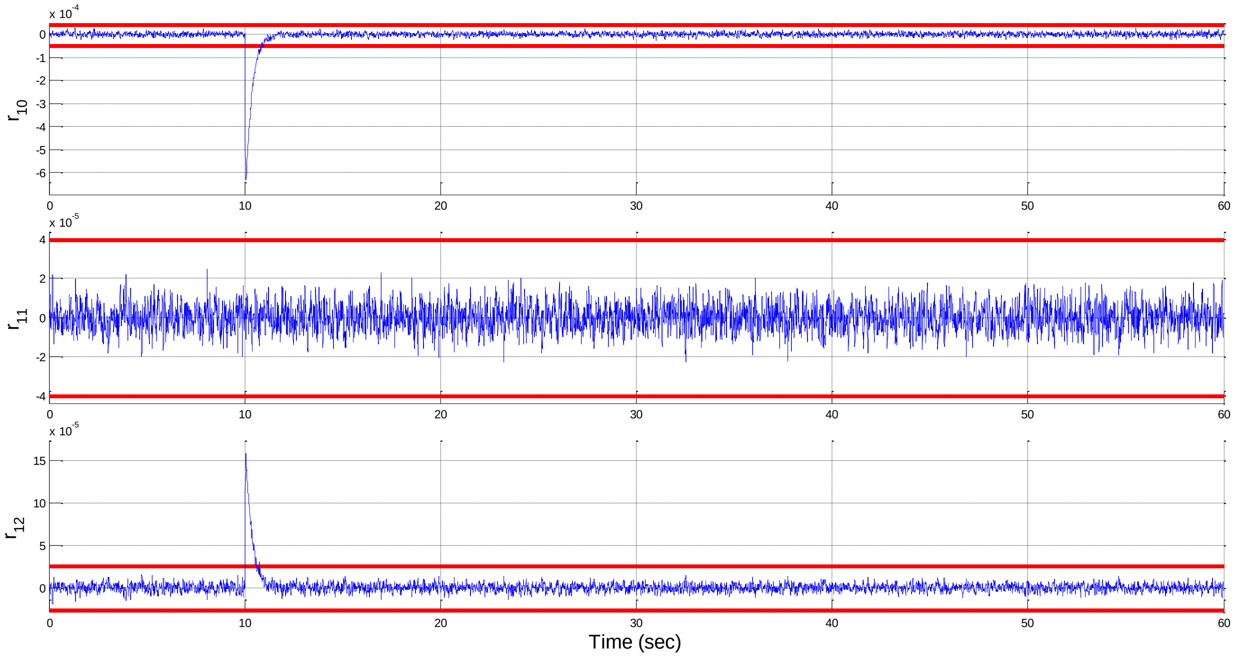


Figure 4.29 - Fault scenario n.12: residuals of RB3, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\overline{\mathbf{q}}_{star1}$, in case of fault on the first measured attitude quaternion vector.

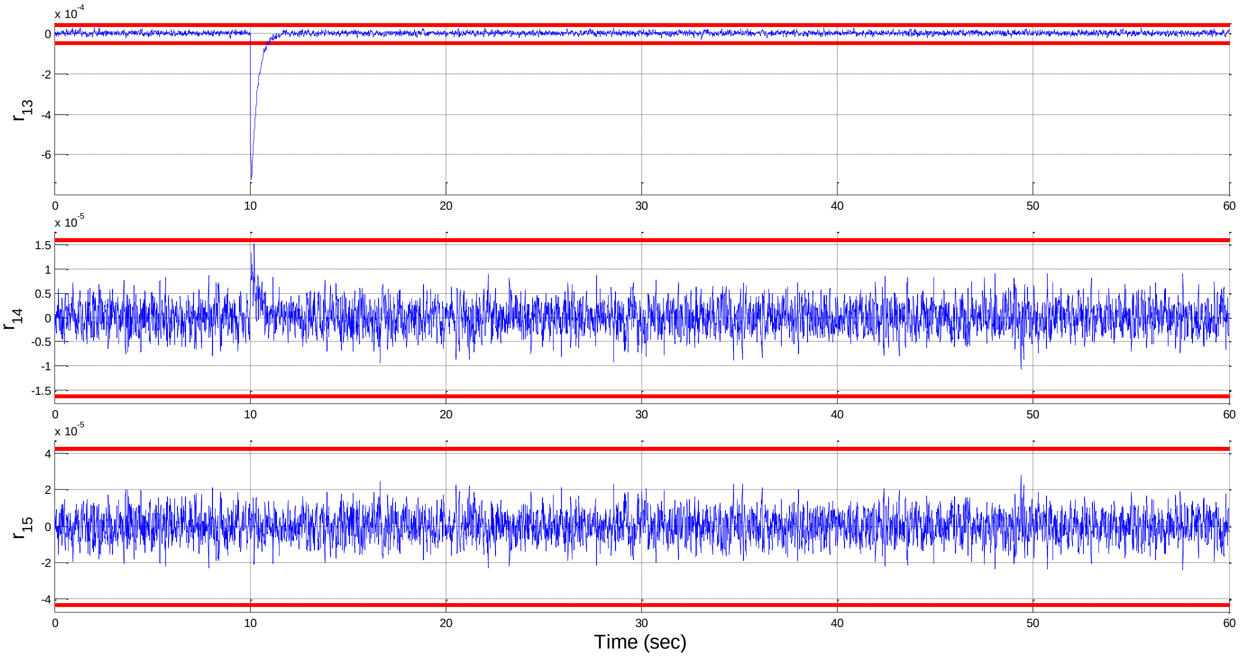


Figure 4.30 - Fault scenario n.12: residuals of RB3, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\overline{\mathbf{q}}_{star1}$, in case of fault on the first measured attitude quaternion vector.

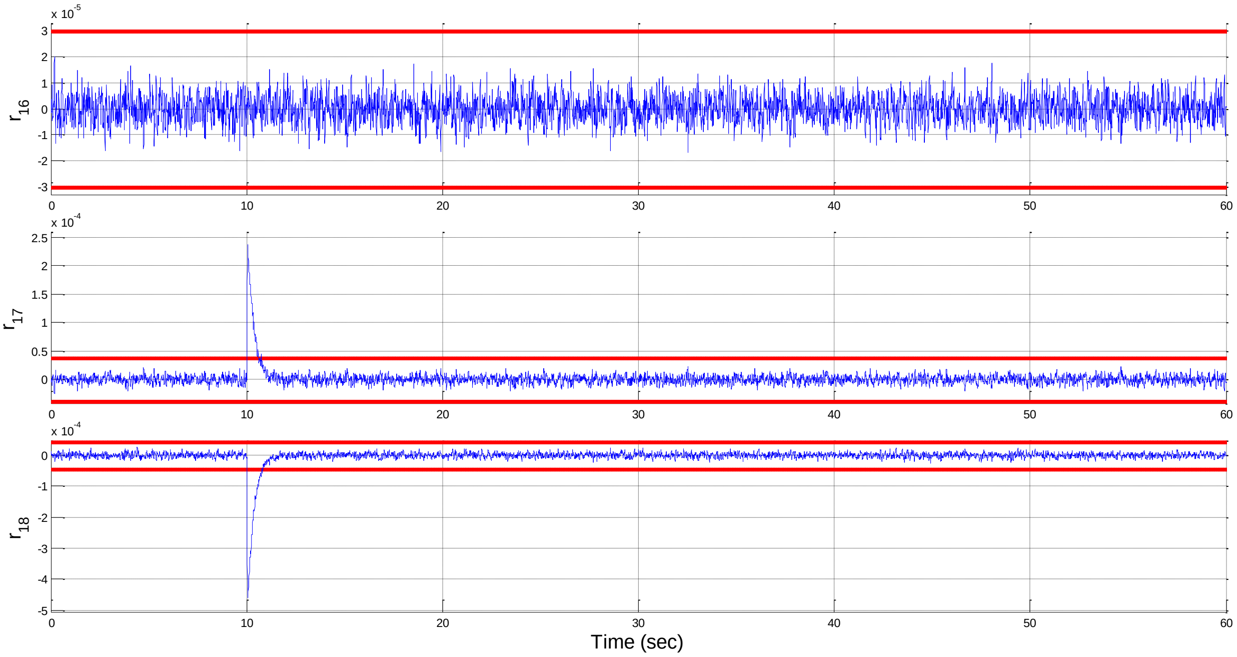


Figure 4.31 - Fault scenario n.12: residuals of RB3, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star1}$, in case of fault on the first measured attitude quaternion vector.

On the contrary, Figs. 4.32, 4.33 and 4.34 show the diagnostic signals provided by RB4. These signals results to be not sensitive to faults affecting the first attitude sensor. Hence, they do not exceed their thresholds because the nine residual filters of RB4 are sensitive to faults affecting the attitude measurements provided by the second attitude sensor only.

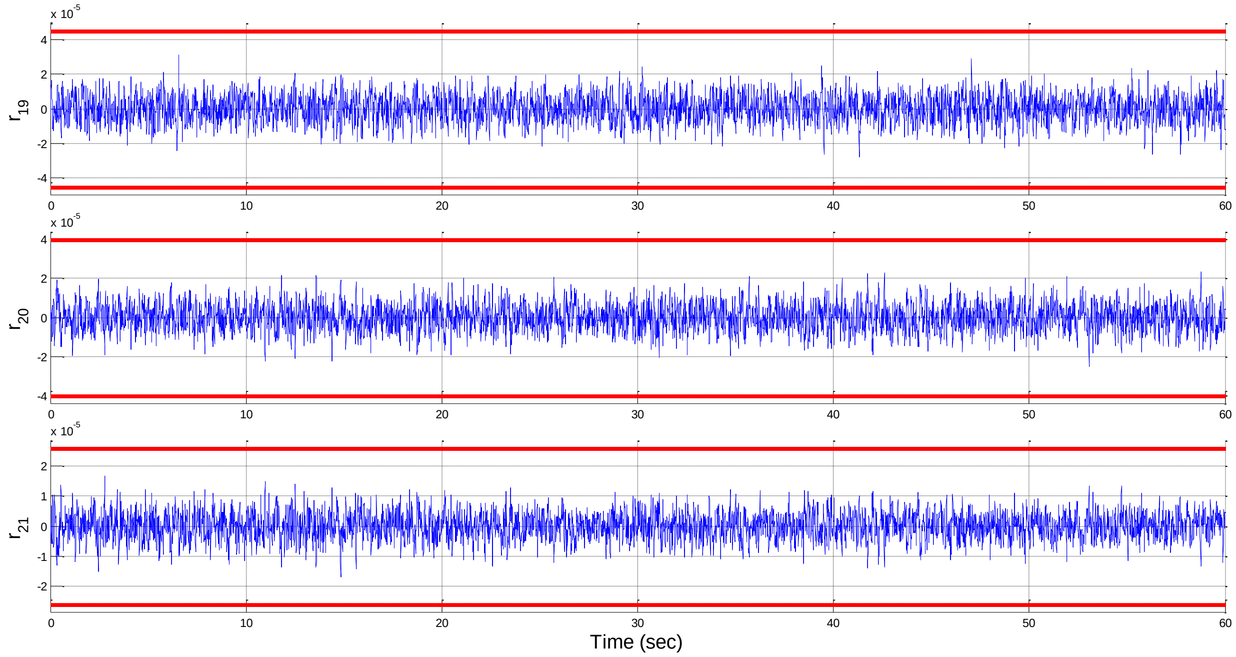


Figure 4.32 - Fault scenario n.12: residuals of RB4, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{x,i}$, $\omega_{y,i}$ and $\bar{\mathbf{q}}_{star2}$, in case of fault on the first measured attitude quaternion vector.

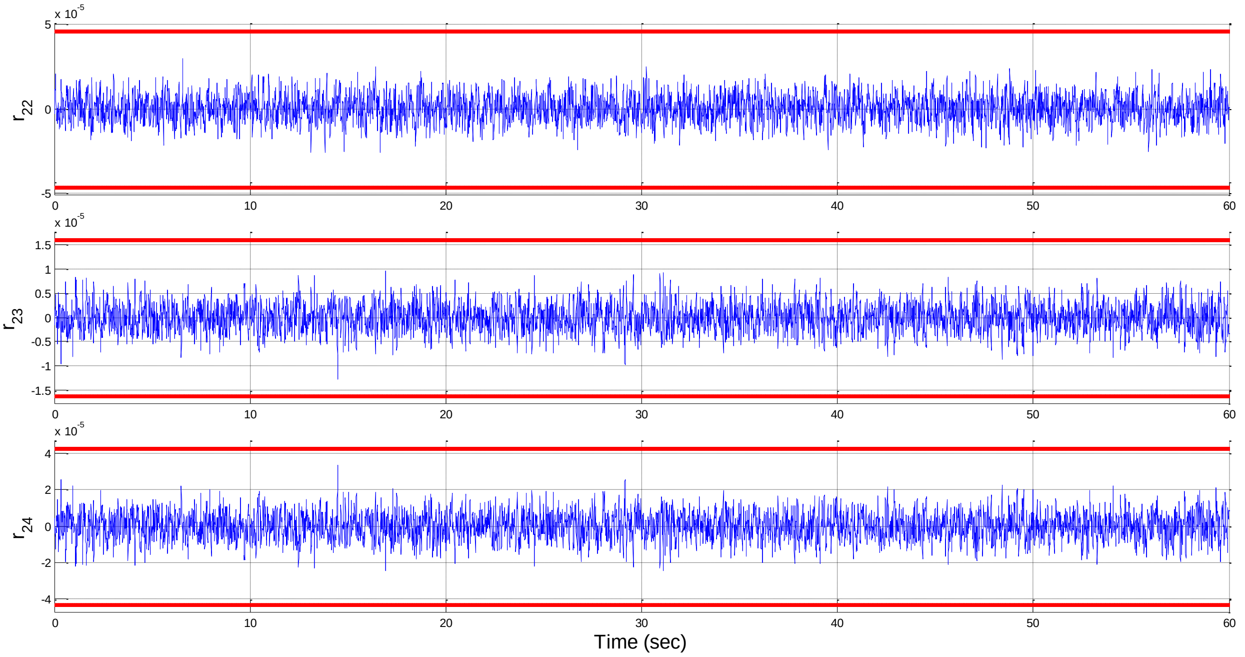


Figure 4.33 - Fault scenario n.12: residuals of RB4, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{x,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star2}$, in case of fault on the first measured attitude quaternion vector.

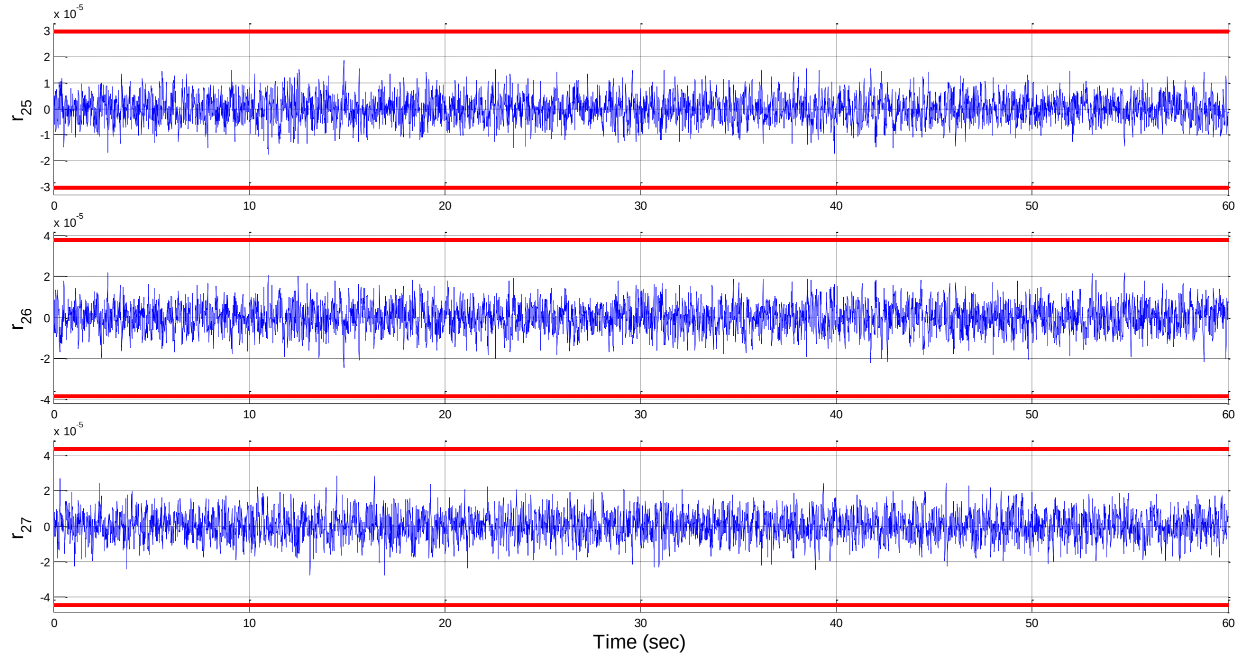


Figure 4.34 - Fault scenario n.12: residuals of RB4, sensitive to fault inputs associated to the physical faults on the sensor measuring $\omega_{y,i}$, $\omega_{z,i}$ and $\bar{\mathbf{q}}_{star2}$, in case of fault on the first measured attitude quaternion vector.

Therefore, from the comparison of these three sets of signals, it is clear that the first attitude sensor measuring $\bar{\mathbf{q}}_{star1}$ is affected by a fault.

5 FAULT DETECTION AND DIAGNOSIS

This chapter provides an overview of the FDD problem, together with the general description of the adaptive fault estimation algorithm exploited in this thesis for the on-line accurate estimation of the occurred fault. The sensor and actuator faults are modelled by means of radial basis function neural networks, allowing designing generalized fault estimation adaptive observers, which do not need any a priori information about the fault internal model. The on-line learning capability of the radial basis function neural network allows obtaining accurate adaptive estimates of the occurred faults.

5.1 General Overview

In Chapter 3, the terms fault detection, fault isolation, and fault identification were introduced. As stated, fault detection is absolutely necessary for any practical application. Fault identification, on the other hand, whilst undoubtedly helpful, may not be essential if no controller redesign is involved. In Active Fault Tolerant Control (AFTC) however, the fault feature is one of the most critical information required for controller redesign (Patton 1997a; Zhang and Jiang 2008). Hence, the fault identification/estimation must be considered in an AFTC process, in which not only the fault alarm and location of the faults are required, but also the time response characteristic of each fault should be known.

The term Fault Detection and Diagnosis (FDD) tends to be used totally in the aerospace flight control community, whereas the wider control-based fault diagnosis community tend to use the term Fault Detection and Isolation (FDI). However, it should be clear the difference between these two terms. In particular, FDI does not include fault reconstruction/estimation/identification and this should be reserved for the topic of FDD.

A common purpose of the FDD functions is to serve either individually or in combination with others for the actuation of a particular fault accommodation scheme, to detect, isolate and estimate faults so that this information can be used in a fault accommodation scheme (AFTC) (Zhang and Jiang 2008). The FDD system provides descriptive information to reconstruct a fault in the form of a signal and notify the AFTC which accommodates the faults by adaptively controlling system dynamics or reconfiguring system structure. A real-time Fault Estimation (FE) function is nowadays regarded as a must-have feature as AFTC schemes require accurate fault information. In this thesis a FE approach is used to provide fault information for fault accommodation within AFTC schemes.

The topic of FE has become well accepted in the control community based on various robust estimation approaches (Edwards *et al.* 2000; Jiang *et al.* 2004; Gao *et al.* 2007). If a fault can be perfectly (or accurately) estimated, all the information including type, size, location and time of occurring can be obtained. Fig. 5.1 shows the structure of FE-based FDD design, in which the FDD function often makes use of FE-based FDD design rather than residual-based FDI design.

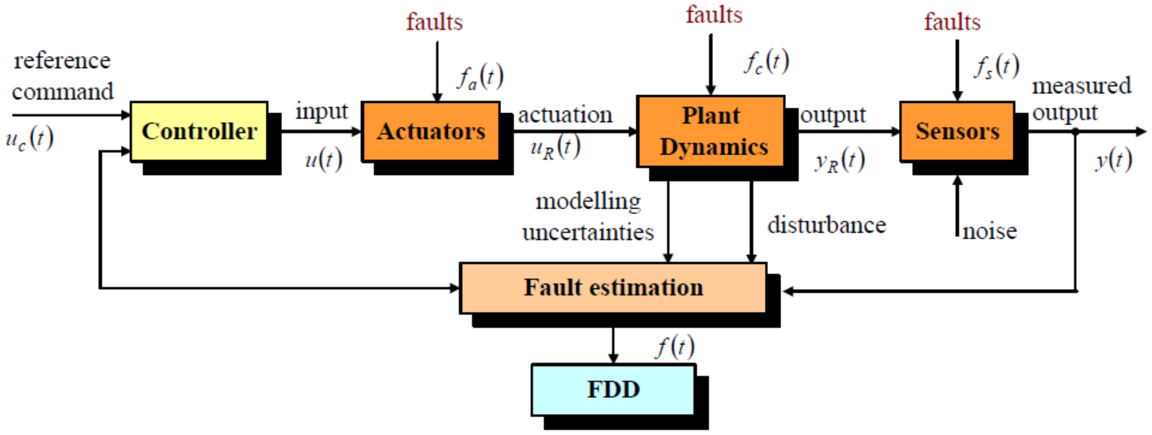


Figure 5.1 - Schematic description of model-based FDD method (Sun 2013).

In general, the determination of accurate fault signal values has become an attractive subject since FE gives a more direct way to achieve the fault information (detection and isolation) than the alternative use of fault indicator or residual signals. The FE approach is a more direct way to obtain this information and the fault estimates can be used directly within some AFTC strategies, for example using fault-hiding and fault compensation (Blanke *et al.* 2006; Wu *et al.* 2006; Zhang 2009; Ponsart *et al.* 2010; Nazari *et al.* 2013). The advantage of model based FDD is to make full use of the information of the process. Nevertheless, the FE problem is always accompanied by some estimation uncertainty, which can be minimized according to a suitable robustness performance. Thus, the performance would heavily depend on the accuracy of the system model, specifically for the quantitative model-based approaches. In other words, FE can be used to achieve fault detection, isolation and identification in one step instead of two or three steps, but it is essential that the robustness problem is solved correctly.

Similarly to FDI problems, since it is not possible to obtain a perfect model as the systems are nonlinear in nature and there is often disturbance or system uncertainty, the mismatch between the mathematic model and real system may cause some problems and enough attention should be paid. For instance, the mismatch may cause false alarms and deteriorate the performance of the system to some extent that the FDD/FE system may even become totally useless. Therefore, it could be a key issue to design a robust FDD/FE system, which is insensitive to unknown inputs such as disturbances, noises and model uncertainty.

Observer designs have received increasing attention in the literature due to the availability of powerful control theory tools. The principle underling observer design is used to estimates of the actual system output (Frank 1987). The output estimation error is obtained by comparing the estimated system outputs and their measured or expected values. Consequently, the residuals or FE signals are designed as a function of the output estimation error.

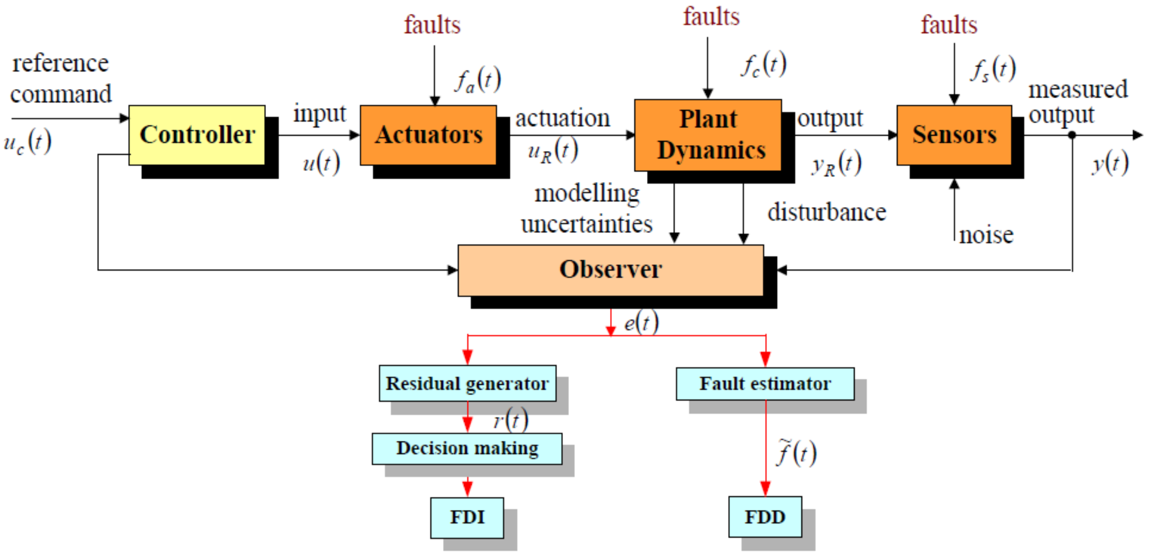


Figure 5.2 - Observer-based FDI/FDD schemes (Sun 2013).

5.2 Fault Diagnosis Based on NLGA and Neural Networks

In this section, the development of an adaptive fault estimation filter/observer for an affine nonlinear system, providing fault size estimation based on a neural network, is presented. Moreover, the uniformly ultimately boundedness property of the closed-loop system error of the fault estimator is formally proven.

5.2.1 Radial Basis Function Neural Networks

Radial Basis Function Neural Networks (RBF-NN)s are a type of feed-forward neural networks. They are used in a wide variety of contexts such as function approximation, pattern recognition, and time series prediction (Park and Sandberg 1991; Chen and Chen 1995; Buhmann 2003). Networks of this type have the universal approximation property. The standard RBF neural network consists of three layers: an input layer, a hidden layer with N neurons, and an output layer. Fig. 5.3 shows a schematic representation of the RBF neural network. The number of the nodes in the input and output layers is decided by the research objects. The nodes in the input layer and output layer represent the vector from an input space and a desired network response, respectively. Through a defined learning algorithm, the error between the actual and desired response can be minimized by optimization criterions.
Each of the input nodes is connected to all the nodes or neurons in the hidden layer through unity weights (direct connections). While each of the hidden layer nodes is connected to the output node through some weights, for example, the *i-th* output node is connected with all the hidden layer nodes by $W_i(t) = \left[w_{i,1}(t), \ldots, w_{i,N}(t) \right]^T$, each neuron finds the distance, normally applying Euclidean norm, between the input and its centre and passes the resulting scalar through a nonlinearity. So, as depicted in Fig. 5.3, the *i-th* output node of the RBF network can be expressed as follows:

$$y_i = \sum_{k=1}^{N} w_{i,k} \varphi_k (\|x - \mu_k\|) \tag{5.1}$$

where $x = [x_1, x_2, \ldots, x_n]^T$ is an input value, n is the number of input node, c_k is the centre of the *k-th* RBF node in the hidden layer, $k = 1, 2, \ldots, N$, and N is the number of hidden nodes. $\|x - \mu_k\|$ denotes Euclidean distance between μ_k and x, φ_k is the nonlinear transfer function of the *k-th* RBF node, $w_{i,k}$ is the weighting value between the *k-th* RBF node and the *i-th* output node and m is the number of output nodes.

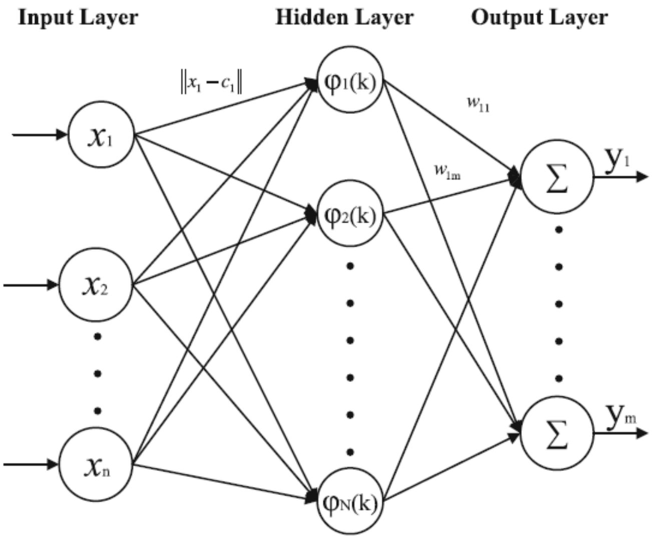


Figure 5.3 - Schematic representation of RBF neural network.

Eq. (5.1) reveals that the output of the network is computed as a weighted sum of the hidden layer outputs. The nonlinear output of the hidden layer is described as φ_k, which is radial symmetrical. Different types of radial basis functions could be used, but the most common is the Gaussian function. In this thesis, the function chosen as activation function for the neural network is the Gaussian function, defined as follows:

$$\varphi_k(x) = \exp\left(-\frac{|x - \mu_k|^2}{\delta_k^2} \right) \tag{5.2}$$

where μ_k and δ_k are the parameters of position and width of the *k-th* RBF node respectively. In fact, the activation function most commonly used for classification and regression problems is the Gaussian function, because it is continuous, differentiable, it provides a softer output and improves the interpolation capabilities. The general procedure to design an RBF neural network for functional approximation problem is shown below:

1. Initialize number N of the RBF nodes;
2. Initialize position μ_k of the RBF nodes;
3. Initialize the width δ_k of the RBF nodes;
4. Calculate the optimum value for the weights $w_{i,k}$;
5. In case, it could be also possible to apply local search algorithms to adjust the parameters of the radial basis functions (*i.e.* the number, widths and positions of the RBF nodes and the corresponding weights) for self-organizing neural networks with adaptive procedures, in order to achieve the best estimation performances in any operative and fault condition.

5.2.2 Adaptive RBF-NN Filter for Generic Fault Estimation

The RBF Neural Network is exploited to model a generic fault function by means of Radial Basis Functions (RBF)s in adaptive fault estimation filters/observers (Wang *et al.* 2011; Baldi *et al.* 2013, 2014a, 2015; Castaldi *et al.* 2014). The on-line learning capability of the Radial Basis Function

Neural Network allows obtaining accurate adaptive estimates of the occurred faults. Moreover, the use of a Radial Basis Function Neural Network allows designing generalized fault estimation adaptive observers which do not need any a priori information about the fault internal model. The generic scalar nonlinear model of the affine form

$$\begin{cases} \dot{x} = n(x) + g(x)u + \ell(x)f \\ y = h(x) \end{cases} \tag{5.3}$$

with initial condition $x(0) = x_0$ is considered. For a sufficiently large number N of hidden-layers neurons and if the system state x takes on values in a sufficiently large compact set $X \subset \mathbb{R}^n$, a weight matrix W can be determined such that a continuous fault function f is approximated by the RBF-NN and the generic fault function f can be linearly parameterized, with a guaranteed finite model error e_m^*, as follows (Chen and Chen 1995):

$$\begin{gathered} f(x) = W^*\varphi(x) + e_m(x) = \sum_{k=1}^{N} w_k^*\varphi_k(x) + e_m(x), \quad |e_m(x)| \le e_m^*, \quad x \in X \\ |f(x) - W^*\varphi(x)| = \left| f(x) - \sum_{k=1}^{N} w_k^*\varphi_k(x) \right| = |e_m(x)| \le e_m^* \end{gathered} \tag{5.4}$$

where $W^* \in \mathbb{R}^N$ is an (unknown) optimal constant weight matrix, *i.e.* $\dot{W}^* = 0$, $\varphi(x)$ is a known vector of basis functions of the form $\varphi(x) = \left[\varphi_1(x), \varphi_2(x), ..., \varphi_N(x)\right]^T \in \mathbb{R}^N$, φ_k is *k-th* radial basis function and e is an approximation error (residual error) between the actual fault function and its optimal RBF approximation, due to the number and type of the selected RBFs.
The general model of an adaptive fault estimator, based on a RBF-NN for a system affine in the fault inputs as Eq. (5.3), can be defined as follows in a filter form (Benini *et al.* 2008):

$$\begin{cases} \dot{\xi} = n_{1s}(\bar{y}_1, \bar{y}_2) + g_{1s}(\bar{y}_1, \bar{y}_2)u + \ell_{1s}(\bar{y}_1, \bar{y}_2)\hat{f} + K(\bar{y}_{1s} - \xi) \\ \varepsilon = \bar{y}_{1s} - \xi \end{cases} \tag{5.5}$$

or, in an alternative way, in an observer form (Benini *et al.* 2008):

$$\begin{cases} \dot{\xi} = n_{1s}(\xi, \bar{y}_{1c}, \bar{y}_2) + g_{1s}(\xi, \bar{y}_{1c}, \bar{y}_2)u + \ell_{1s}(\xi, \bar{y}_{1c}, \bar{y}_2)\hat{f} + K(\bar{y}_{1s} - \xi) \\ \varepsilon = \bar{y}_{1s} - \xi \end{cases} \tag{5.6}$$

where $K > 0$ represents the filter/observer gain, which can be designed such that the residual generator of Eq. (5.5) is asymptotically stable with a good fault sensibility versus noise attenuation ratio. The unknown fault function f is linearly parameterized in Eq. (5.5) by a RBF-NN:

$$\hat{f} = \hat{W}\varphi(\xi) \tag{5.7}$$

with the weight matrix $\hat{W} \in \mathbb{R}^N$ determined by the following adaptive weight update law:

$$\dot{\hat{W}} = \eta D \varepsilon \varphi^T(\xi) \tag{5.8}$$

where $\eta > 0$ is the learning ratio (positive fixed gain) and D is a proper constant matrix such that the RBF-NN adaptive filter composed by Eq. (5.5) with the fault estimate of Eq. (5.7) and the weight adaptive law of Eq. (5.8) is asymptotically stable and the fault estimate converges to the real fault value. The residual ε which drives the adaptive law of the RBF weight matrix (and thus the behaviour of the fault estimate) returns to zero when the obtained fault estimate $\hat{f}$ converges to the actual fault f, and consequently the weight adaptation stops.

5.2.3 Estimation Error Boundedness

The estimation error can be defined as the difference between the true exogenous disturbance and its estimation:

$$\begin{aligned} e_{est}(t,x) &= f(t,x) - \hat{W}(t)\varphi(x(t)) = \\ &= f(t,x) - W(t)\varphi(x(t)) + W(t)\varphi(x(t)) - \hat{W}(t)\varphi(x(t)) = \\ &= e_m + \tilde{W}(t)\varphi(x(t)) \end{aligned} \tag{5.9}$$

where the term $\tilde{W}(t)\varphi(x(t))$ represents the adaptation error, *i.e.* the difference between the optimal and the estimated RBF approximation. The weight error vector is defined as $\tilde{W}(t) = W(t) - \hat{W}(t)$.
The estimation error e_{est} can be proven to be bounded (Castaldi *et al.* 2014). The origin of the system composed by the detection filter of Eq. (5.5) and the adaptive law of Eq. (5.8) is globally asymptotically stable (0-GAS) with respect to the input e_m if there exist a matrix D and a positive definite matrix P such that

$$\begin{aligned} \ell_{1s}^T P &= D \\ PK &> 0 \end{aligned}$$

In fact, starting by defining the following function

$$V = \frac{1}{2} P\varepsilon^2 + \frac{1}{2\eta} \tilde{W}\tilde{W}^T > 0 \quad \forall \varepsilon, \tilde{W} > 0 \tag{5.10}$$

its time derivative is hereafter obtained, considering an optimal constant weight matrix, as

$$\dot{V} = \varepsilon P\dot{\varepsilon} + \frac{1}{\eta}\sum_{k=1}^{N} \tilde{w}_k \dot{\tilde{w}}_k = \varepsilon P\left[(f - \hat{f}) - K\varepsilon\right] - \frac{1}{\eta}\sum_{k=1}^{N} \tilde{w}_k D\varepsilon\varphi_k(\xi) = -\varepsilon^2 PK + \varepsilon(P - D)\sum_{k=1}^{N} \tilde{w}_k \varphi_k(\xi)$$

If $P = D$ and $PK > 0$, it follows that

$$\dot{V} = -\varepsilon^2 PK < 0 \quad \forall \varepsilon > 0 \tag{5.11}$$

The Lyapunov function derivative $\dot{V}$ is only semi-definite negative, *i.e.* $\dot{V}(\varepsilon, \tilde{W}) = 0$ for $\varepsilon = 0$ and $\forall \tilde{W} \neq 0$. This implies that only the filter error (residual) ε goes to zero whilst $\tilde{W}$ can be finite and different from zero. The LaSalle's Principle can be applied to deduce that also $\tilde{W}(t)$ goes to the origin if $t \to \infty$. Taken the bounded set

$$\Omega_C = \left\{ x = (\varepsilon, \tilde{W}) \in \mathbb{R}^{N+1} \,|\, V(x) \leq C \in \mathbb{R}^+ \right\}$$

such that $\dot{V} \leq 0$ for all $x \in \Omega_C$, defining $S \subset \Omega_C$ by

$$S = \left\{ x \in \Omega_C \mid \dot{V}(x) = 0 \right\}$$

and with M the largest invariant set in S, then, whenever $x_0 \in \Omega_C$, all the trajectories starting from x_0 approach to M as $t \to \infty$. In this case the invariant set M is composed by only one equilibrium point, that is the origin $(\varepsilon, \tilde{\mathrm{W}}) = (0,0)$.

Starting from the these results, it is straightforward to see that the estimation error is bounded. Thanks to the Local ISS property of the system (Sontag, Wang 1996), Lemma I.1) the weight error vector is bounded, *i.e.* there exist two functions β (KL-class) and γ (K-class) such that

$$\|\tilde{W}\| \le \beta(\|\tilde{W}(0)\|, t) + \gamma(|e_m|_\infty) = \|\tilde{W}^*\|$$
$$\|\tilde{W}(0)\| < \rho \in \mathbb{R}^+$$
$$|e_m|_\infty < d \in \mathbb{R}^+$$

from which it results

$$|e_{est}| = |e_m + \tilde{W}\varphi(x)| \le \varepsilon_m + N\|\tilde{W}^*\| \tag{5.12}$$

6 FDD IMPLEMENTATION IN THE SATELLITE ADCS

The overall satellite FDD system consists of the previously described FDI module and of a FE module, whose outputs (*i.e.* the fault estimates) are enabled only once an occurred fault has been accurately detected and isolated by the FDI module, *i.e.* the four banks of designed residual filters and the relative residual cross-checking procedure. This fault estimation module consists of thirteen independent adaptive fault estimators, each of them responsible of providing the estimate of a specific fault affecting one of the actuated attitude control torques or sensor measurements.

These adaptive fault estimators are derived from the model equations (4.5) and (4.7), and from the derived residual filters and they are exploited in order to accurately estimate the size of the occurred faults, by exploiting Radial Basis Function Neural Networks (RBF-NN)s. As it will be shown, with the assumption of a single fault at a time, the use of these adaptive fault estimators allows obtaining very accurate fault estimates.

This chapter presents the development of the nonlinear FE adaptive filters providing the estimation of the fault sizes. The information brought by the fault size estimations can be very useful for off-line maintenance purposes and for on-line reconfiguration of the automatic control system. The proposed fault estimation algorithm is based on the NLGA and Radial Basis Function Neural Networks (RBF-NN)s.

6.1 Design of Adaptive RBF-NN Filters for Fault Estimation in the Satellite ACS

Thanks to the NLGA procedure, scalar $\bar{x}_{1s}$-subsystems affine in the actuator fault inputs have been derived from the spacecraft nonlinear model in Chapter 4. Considering the components of the spacecraft ACS, the adaptive filters for the estimation of faults affecting the control torques $\mathbf{T}_{ctrl} = \begin{bmatrix} T_{ctrl,1} & T_{ctrl,2} & T_{ctrl,3} & T_{ctrl,4} \end{bmatrix}^T$ are derived from the models of the local residual filters of RB1, which have been designed in Chapter 4. As previously stated, each of these local residual filters is sensitive only to faults affecting a single actuator subsystem, and hence a single control torque since they are based on the dynamic equations of the actuators.

On the basis of the adaptive filter model of Eq. (5.5), the general model of Eq. (4.36) of these scalar local residual filters for the $\bar{x}_{1s}$-subsystems can be modified, in order to obtain adaptive filters based on a RBF-NN for the estimation of faults affecting the attitude control inputs $\mathbf{T}_{ctrl}$, as follows:

$$\begin{cases} \dot{\xi} = n_{1s}(\bar{y}_1, \bar{y}_2) + g_{1s}(\bar{y}_1, \bar{y}_2)u + \ell_{1s,u}(\bar{y}_1, \bar{y}_2)\,\hat{f}_u + K(\bar{y}_{1s} - \xi) \\ \varepsilon = \bar{y}_{1s} - \xi \end{cases} \tag{6.1}$$

where $K > 0$ and the unknown fault function f_u affecting the control torque is linearly parameterized in Eq. (6.1) by a RBF-NN:

$$\hat{f} = \hat{W}\varphi(\xi) \tag{6.2}$$

with the weight vector $\hat{W} \in \mathbb{R}^N$ determined by the following adaptive weight update law:

$$\dot{\hat{W}} = \eta D\varepsilon\varphi^T(\xi) = \eta D\varepsilon\varphi^T(\xi) \tag{6.3}$$

where $\eta > 0$ is the learning ratio, *D* is a proper constant value such that the RBF-NN adaptive filter with model defined by Eqs. (6.1), (6.2) and (6.3) is asymptotically stable.

The selected value *D* of the adaptive filters should guarantee good convergence speeds of the fault estimates together with a sensitivity to the noise as low as possible. The residual ε which drives the adaptive law of the RBF weight matrix (and thus the behaviour of the fault estimate) returns to zero when the obtained fault estimate $\hat{f}_u$ converges to the actual actuator fault f_u.

The assumption of single fault occurring at a time and the exploitation of the local residual filter models allows to design RBF-NN adaptive fault estimation filters which directly provide the estimates $\hat{f}_u$ of the possible faults $f_u = T_{ctrl} - T_c$ affecting a specific actuated control torques. The output of the RBF-NN adaptive estimation filter is activated once the corresponding actuator fault has been isolated by the FDI module described in Chapter 4.

Starting from Eqs. (4.5), (6.1), (6.2) and (6.3), it results that the models of the scalar adaptive estimation filters for faults affecting the four actuated attitude control torques considered in this thesis can be written as follows:

$$\begin{cases} \dot{\xi}_{u_1} = -\frac{T_{c,1}}{I_w} - \frac{\hat{f}_{u_1}}{I_w} + K(\omega_{w,1_{measured}} - \xi_{u_1}) \\ \varepsilon_{u_1} = \omega_{w,1_{measured}} - \xi_{u_1} \\ \hat{f}_{u_1} = \hat{W}\varphi(\xi_{u_1}) \\ \dot{\hat{W}} = \eta D \varepsilon_{u_1} \varphi^T(\xi_{u_1}) \end{cases} \tag{6.4}$$

$$\begin{cases} \dot{\xi}_{u_2} = -\frac{T_{c,2}}{I_w} - \frac{\hat{f}_{u_2}}{I_w} + K(\omega_{w,2_{measured}} - \xi_{u_2}) \\ \varepsilon_{u_2} = \omega_{w,2_{measured}} - \xi_{u_2} \\ \hat{f}_{u_2} = \hat{W}\varphi(\xi_{u_2}) \\ \dot{\hat{W}} = \eta D \varepsilon_{u_2} \varphi^T(\xi_{u_2}) \end{cases} \tag{6.5}$$

$$\begin{cases} \dot{\xi}_{u_3} = -\frac{T_{c,3}}{I_w} - \frac{\hat{f}_{u_3}}{I_w} + K(\omega_{w,3_{measured}} - \xi_{u_3}) \\ \varepsilon_{u_3} = \omega_{w,3_{measured}} - \xi_{u_3} \\ \hat{f}_{u_3} = \hat{W}\varphi(\xi_{u_3}) \\ \dot{\hat{W}} = \eta D \varepsilon_{u_3} \varphi^T(\xi_{u_3}) \end{cases} \tag{6.6}$$

$$\begin{cases} \dot{\xi}_{u_4} = -\frac{T_{c,4}}{I_w} - \frac{\hat{f}_{u_4}}{I_w} + K(\omega_{w,4_{measured}} - \xi_{u_4}) \\ \varepsilon_{u_4} = \omega_{w,4_{measured}} - \xi_{u_4} \\ \hat{f}_{u_4} = \hat{W}\varphi(\xi_{u_4}) \\ \dot{\hat{W}} = \eta D \varepsilon_{u_4} \varphi^T(\xi_{u_4}) \end{cases} \tag{6.7}$$

In particular, for the adaptive filters described above, *N=11* Gaussian RBFs have been considered, with width $\sigma_k = 200$ and centres μ_k equally spaced between -200 rad/s and 200 rad/s (±1909 rpm).

On the other hand, once a fault on one of the four flywheel spin rate sensors has been correctly detected and isolated by the FDI module, the output of the corresponding RBF-NN adaptive fault estimation filter is enabled. Again, the models of the four local residual filters of RB1 are exploited in order to design four distinct RBF-NN adaptive fault estimation filters. However, in this case these adaptive filters have not been designed exactly on the basis of the adaptive filter model of Eq. (5.5), but on the basis of a slightly modified version.

This is made in order to allow the direct and accurate estimation $\hat{F}_{\omega_w}$ of the actual fault affecting the flywheel spin rate sensor. Therefore, the general model of Eq. (4.36) of the scalar local residual filters has been modified, in order to obtain adaptive filters for a direct sensor fault estimation based on a RBF-NN, as follows:

$$\begin{cases} \dot{\xi} = n_{1s}(\bar{y}_1, \bar{y}_2) + g_{1s}(\bar{y}_1, \bar{y}_2)u + K(\bar{y}_{1s} - \xi - \hat{F}) \\ \varepsilon = \bar{y}_{1s} - \xi - \hat{F} \end{cases} \tag{6.8}$$

where the unknown fault function $\hat{F}$ is linearly parameterized by a RBF-NN:

$$\hat{F} = \hat{W}\varphi(\xi) \tag{6.9}$$

with the weight vector $\hat{W} \in \mathbb{R}^N$ determined by the following adaptive law:

$$\dot{\hat{W}} = \eta D\varepsilon\varphi^T(\xi) = \eta D\varepsilon\varphi^T(\xi) \tag{6.10}$$

where $\eta > 0$ is the learning ratio, *D* is a proper constant value such that the RBF-NN adaptive filter defined by Eqs. (6.8), (6.9) and (6.10) is asymptotically.

In this case, in Eq. (6.8) the filter model has been modified in a different way with respect to Eq. (5.5) in order to obtain RBF-NN adaptive fault estimation filters whose outputs directly provide accurate flywheel spin rate sensor fault estimates. In fact, in order to bring back to zero after the fault occurrence the residual ε which drives the adaptive law of the RBF weight matrix (and thus the behaviour of the fault estimate), the estimation $\hat{F}_{\omega_w}$ of a sensor fault is directly added to the value of the state variable ξ of the filter in Eq. (6.8). In this way, it is obtained the output variable of the filter $y_\xi = \xi + \hat{F}_{\omega_w}$ to be compared with the measured output of the real satellite system affected by the actual sensor fault F_{ω_w}. When the obtained fault estimate $\hat{F}_{\omega_w}$ converges to the actual sensor fault F_{ω_w}, the residual ε returns to zero and the adaptation of the weights of the radial basis functions stops.

Again, the assumption of single fault occurring at a time and the exploitation of the local residual filter models allows to design RBF-NN adaptive fault estimation filters which provide directly the estimates $\hat{F}_{\omega_w}$ of the possible faults $F_{\omega_w} = \omega_{w_{measured}} - \omega_{w_{true}}$ affecting a specific flywheel spin rate sensor. In fact, each of the local residual filters of RB1 results to be sensitive also to the faults affecting a specific flywheel spin rate sensor (in addition to the faults affecting the corresponding actuated control torque) with unitary gain.

Starting from Eqs. (4.5), (6.8), (6.9) and (6.10), it results that the models of the four scalar adaptive estimation filters for faults affecting the flywheel spin rate sensors can be written as follows:

$$\begin{cases} \dot{\xi}_{\omega_{w,1}} = -\dfrac{T_{c,1}}{I_w} + K(\omega_{w,1_{measured}} - \xi_{\omega_{w,1}} - \hat{F}_{\omega_{w,1}}) \\ \varepsilon_{\omega_{w,1}} = \omega_{w,1_{measured}} - \xi_{\omega_{w,1}} - \hat{F}_{\omega_{w,1}} \\ \hat{F}_{\omega_{w,1}} = \hat{W}\varphi(\xi_{\omega_{w,1}}) \\ \dot{\hat{W}} = \eta D \varepsilon_{\omega_{w,1}} \varphi^T(\xi_{\omega_{w,1}}) \end{cases} \tag{6.11}$$

$$\begin{cases} \dot{\xi}_{\omega_{w,2}} = -\dfrac{T_{c,2}}{I_w} + K(\omega_{w,2_{measured}} - \xi_{\omega_{w,2}} - \hat{F}_{\omega_{w,2}}) \\ \varepsilon_{\omega_{w,2}} = \omega_{w,2_{measured}} - \xi_{\omega_{w,2}} - \hat{F}_{\omega_{w,2}} \\ \hat{F}_{\omega_{w,2}} = \hat{W}\varphi(\xi_{\omega_{w,2}}) \\ \dot{\hat{W}} = \eta D \varepsilon_{\omega_{w,2}} \varphi^T(\xi_{\omega_{w,2}}) \end{cases} \tag{6.12}$$

$$\begin{cases} \dot{\xi}_{\omega_{w,3}} = -\dfrac{T_{c,3}}{I_w} + K(\omega_{w,3_{measured}} - \xi_{\omega_{w,3}} - \hat{F}_{\omega_{w,3}}) \\ \varepsilon_{\omega_{w,3}} = \omega_{w,3_{measured}} - \xi_{\omega_{w,3}} - \hat{F}_{\omega_{w,3}} \\ \hat{F}_{\omega_{w,3}} = \hat{W}\varphi(\xi_{\omega_{w,3}}) \\ \dot{\hat{W}} = \eta D \varepsilon_{\omega_{w,3}} \varphi^T(\xi_{\omega_{w,3}}) \end{cases} \tag{6.13}$$

$$\begin{cases} \dot{\xi}_{\omega_{w,4}} = -\dfrac{T_{c,4}}{I_w} + K(\omega_{w,4_{measured}} - \xi_{\omega_{w,4}} - \hat{F}_{\omega_{w,4}}) \\ \varepsilon_{\omega_{w,4}} = \omega_{w,4_{measured}} - \xi_{\omega_{w,4}} - \hat{F}_{\omega_{w,4}} \\ \hat{F}_{\omega_{w,4}} = \hat{W}\varphi(\xi_{\omega_{w,4}}) \\ \dot{\hat{W}} = \eta D \varepsilon_{\omega_{w,4}} \varphi^T(\xi_{\omega_{w,4}}) \end{cases} \tag{6.14}$$

The centres and widths of the Gaussian functions are assumed a priori and not time-varying since the inputs x can assume values only in well-defined intervals on the basis of the models of the designed adaptive filters and characteristics of the system components. In particular, for the adaptive filters described above, *N=11* Gaussian RBFs have been considered, with width $\sigma_k = 200$ and centres μ_k equally spaced between -200 rad/s and 200 rad/s (±1909 rpm).

6.2 Design of Adaptive RBF-NN Filters for Fault Estimation in the Satellite ADS

Considering the sensors composing the ADS of the spacecraft, once a fault on one of the three spacecraft angular velocity sensors has been correctly detected and isolated by the FDI module, the output of the corresponding RBF-NN adaptive fault estimation filter is enabled. In this case, Eq. (4.7) can be properly exploited in order to design RBF-NN adaptive fault estimation filters. On the basis of the adaptive filter model of Eq. (5.5), adaptive estimation filters can be designed for the mathematical fault inputs associated to each angular velocity state variable comparing in the kinematic equations of the model of Eq. (4.34) affine in the sensor fault inputs. With the assumption of single fault occurring at any time, each kinematic equation in the affine form of Eq. (4.34) with respect to all the fault inputs results to be including a single fault term related to each angular velocity measurement. Then, as the definitions of each associated mathematical fault input is known, the corresponding estimation of the actual sensor fault $\hat{F}_{\omega}$ can be computed once any estimation $\hat{f}_{\omega,k}$ of a proper *k-th* associated mathematical fault input is available from an adaptive filter realized on the basis of the kinematic equations.

Again, a generic adaptive filter based on a RBF-NN for the estimation of an *k-th* associated fault input can be derived on the basis of any kinematic equation, and modelled in filter form as follows:

$$\begin{cases} \dot{\xi} = n_{1s}(\bar{y}_1, \bar{y}_2) + g_{1s}(\bar{y}_1, \bar{y}_2)u + \ell_{1s}(\bar{y}_1, \bar{y}_2)\hat{f} + K(\bar{y}_{1s} - \xi) \\ \varepsilon = \bar{y}_{1s} - \xi \end{cases} \tag{6.15}$$

where the unknown fault function $\hat{f}$ is linearly parameterized by a RBF-NN:

$$\hat{f} = \hat{W}\varphi(\xi) \tag{6.16}$$

with the weight vector $\hat{W} \in \mathbb{R}^N$ determined by the following adaptive law:

$$\dot{\hat{W}} = \eta D\varepsilon\varphi^T(\xi) = \eta D\varepsilon\varphi^T(\xi) \tag{6.17}$$

where $\eta > 0$ is the learning ratio, *D* is a proper constant value such that the RBF-NN adaptive filter defined by Eqs. (6.15), (6.16) and (6.17) is asymptotically stable.

For example, once a fault affecting the spacecraft angular velocity sensor measuring the satellite angular rate $\omega_{x,i}$ in the inertial frame, the following adaptive estimation filter can be exploited to obtain an estimation of the associated mathematical fault input $f_{\omega_{\mathrm{x},i},3} = q_{star1,4}(\omega_{\mathrm{x},i_{measured}} - \omega_{\mathrm{x},i_{true}})$ of Eq. (4.14).

$$\begin{cases} \dot{\xi}_{\omega_{y,i}} = \frac{1}{2}(\omega_{x,i_{measured}} q_{star1,3_{measured}} + \omega_{y,i_{measured}} q_{star1,4_{measured}} - \omega_{z,i_{measured}} q_{star1,1_{measured}} + \omega_o q_{star1,4_{measured}}) + \\ \qquad - \frac{1}{2}\hat{f}_{\omega_{y,i},4} + K(q_{star1,2_{measured}} - \xi_{\omega_{y,i}}) \\ \varepsilon_{\omega_{y,i}} = q_{star1,2_{measured}} - \xi_{\omega_{y,i}} \\ \hat{f}_{\omega_{y,i},4} = \hat{W}\varphi(\xi_{\omega_{y,i}}) \\ \dot{\hat{W}} = \eta D\varepsilon_{\omega_{y,i}}\varphi^T(\xi_{\omega_{y,i}}) \end{cases} \tag{6.18}$$

The corresponding estimation of the actual sensor fault $\hat{F}_{\omega_{y,i}}$ can be computed once any estimation $\hat{f}_{\omega_{\mathrm{x},i},3}$ is available, by using the relation

$$\hat{F}_{\omega_{x,i}} = \frac{\hat{f}_{\omega_{x,i},3}}{q_{star1,4}} = (\omega_{x,i_{measured}} - \omega_{x,i_{true}})$$

when $q_{star1,4} \neq 0$. If the spacecraft attitude is such that the quaternion parameter $q_{star1,4}$ is zero, in order to avoid a singularity in the computation of the actual fault estimation, a different kinematic equation and mathematical relation of a fault input associated to the detected and isolated actual sensor fault can be exploited in the estimation algorithm. For these adaptive filters, *N=11* Gaussian RBFs have been considered, with width $\sigma_k = 1$ and centres μ_k equally spaced between -1 and 1.

Finally, once a fault on one of the two star tracker attitude sensors has been correctly detected and isolated by the FDI module, the output of the corresponding RBF-NN adaptive fault estimation filter is enabled. In this case, however, Eq. (4.7) can be exploited in order to design a RBF-NN adaptive fault estimator for the overall additive fault vector $\bar{\mathbf{F}}_{star}$ affecting the quaternion measurement of $\bar{\mathbf{q}}_{star_{measured}}$, as modelled in Eq. (2.55). Since this fault vector $\bar{\mathbf{F}}_{star}$ is equivalent to $\bar{\mathbf{q}}_f$ but in an additive fault formulation, the corresponding fault estimation $\hat{\bar{\mathbf{F}}}_{star}$ can be directly exploited in an AFTC scheme in order to correct the sensor measurement, *i.e.* compensate the sensor fault and recover an accurate measurement of the actual attitude quaternion vector $\bar{\mathbf{q}}_{star}$.

In this case, the adaptive estimation filter exploits a residual generator model in observer form, and the residual and state vectors of the adaptive filter are vector defined in $\mathbb{R}^4$. Hence, a generic adaptive estimator based on a RBF-NN for the estimation of the sensor fault can be derived as follows:

$$\begin{cases} \dot{\xi} = n_1(\xi, \bar{y}_2) + g_1(\xi, \bar{y}_2)u + K(\bar{y}_1 - \xi - \hat{F}) \\ \varepsilon = \bar{y}_1 - \xi - \hat{F} \end{cases} \tag{6.19}$$

where the unknown fault vector $\hat{F} \in \mathbb{R}^4$ is linearly parameterized by a RBF-NN:

$$\hat{F} = \hat{W}\varphi(\xi) \tag{6.20}$$

with the weight matrix $\hat{W} \in \mathbb{R}^{4\times N}$ determined by the following adaptive law:

$$\dot{\hat{W}} = \eta D\varepsilon\varphi^T(\xi) = \eta D\varepsilon\varphi^T(\xi) \tag{6.21}$$

where $\eta > 0$ is the learning ratio, D is a proper constant matrix such that the RBF-NN adaptive filter defined by Eqs. (6.19), (6.20) and (6.21) is asymptotically stable. On the basis of the four kinematic equations (4.7), the following adaptive estimator in observer form can be exploited to obtain an estimation of the complete fault vector $\hat{\bar{\mathbf{F}}}_{star} = \left[\hat{F}_{star,1}, \hat{F}_{star,2}, \hat{F}_{star,3}, \hat{F}_{star,4}\right]^T$:

$$\begin{cases} \dot{\xi}_{q_{star,1}} = \frac{1}{2}(\omega_{x,i_{measured}}\xi_{q_{star,4}} - \omega_{y,i_{measured}}\xi_{q_{star,3}} + \omega_{z,i_{measured}}\xi_{q_{star,2}} + \omega_o\xi_{q_{star,3}}) + \\ \quad +K(q_{star,1_{measured}} - \xi_{q_{star,1}} - \hat{F}_{star,1}) \\ \dot{\xi}_{q_{star,2}} = \frac{1}{2}(\omega_{x,i_{measured}}\xi_{q_{star,3}} + \omega_{y,i_{measured}}\xi_{q_{star,4}} - \omega_{z,i_{measured}}\xi_{q_{star,1}} + \omega_o\xi_{q_{star,4}}) + \\ \quad +K(q_{star,2_{measured}} - \xi_{q_{star,2}} - \hat{F}_{star,2}) \\ \dot{\xi}_{q_{star,3}} = \frac{1}{2}(-\omega_{x,i_{measured}}\xi_{q_{star,2}} + \omega_{y,i_{measured}}\xi_{q_{star,1}} + \omega_{z,i_{measured}}\xi_{q_{star,4}} - \omega_o\xi_{q_{star,1}}) + \\ \quad +K(q_{star,3_{measured}} - \xi_{q_{star,3}} - \hat{F}_{star,3}) \\ \dot{\xi}_{q_{star,4}} = \frac{1}{2}(-\omega_{x,i_{measured}}\xi_{q_{star,1}} - \omega_{y,i_{measured}}\xi_{q_{star,2}} - \omega_{z,i_{measured}}\xi_{q_{star,3}} + \omega_o\xi_{q_{star,2}}) + \\ \quad +K(q_{star,4_{measured}} - \xi_{q_{star,4}} - \hat{F}_{star,4}) \\ \varepsilon_{q_{star,i}} = q_{star,i_{measured}} - \xi_{q_{star,i}} - \hat{F}_{star,i} \\ \hat{F}_{star,i} = \hat{W}\varphi(\xi_{q_{star,i}}) \\ \dot{\hat{W}} = \eta D\varepsilon_{q_{star,i}}\varphi^T(\xi_{q_{star,i}}) \end{cases} \tag{6.22}$$

For this adaptive estimator, *N=11* Gaussian RBFs have been considered, with width $\sigma_k = 1$ and centres μ_k equally spaced between -1 and 1.

6.3 Simulation Results

The following figures show the fault estimation results obtained by using the designed adaptive fault estimation filters once the occurrence of the faults in the considered fault scenarios have been occurred, detected and correctly isolated. The figures highlight how the designed adaptive estimation filters can achieve an high estimation accuracy in each of the fault scenarios.

6.3.1 Estimation of Faults in the Satellite ACS

Let us start again considering the occurrence of faults affecting the actuators. In the first fault scenario, *i.e.* the step fault, Figure 6.1 shows the effect of the fault on the actuated control torque depicted in blue, with respect to the nominal commanded one depicted in black. The constant bias between the actuated $T_{ctrl,2}$ and commanded inputs $T_{c,2}$ is clearly visible. For the attitude control torque, the maximum saturation value has been set equal to $T_{ctrl,sat} = \pm 0.75$ *Nm* and depicted in red in the figure.

Fig. 6.2 shows the fault estimation obtained by exploiting the designed adaptive fault estimation filters based on the RBF-NN. In this figure, the actual fault is depicted with a red line, whereas the fault estimation is depicted in blue. Moreover, the obtained fault estimation error is shown in Fig. 6.3. It can be seen that the estimation error has a zero mean value, and it is bounded, except for the single error peak appearing at the instant of occurrence of the abrupt fault.

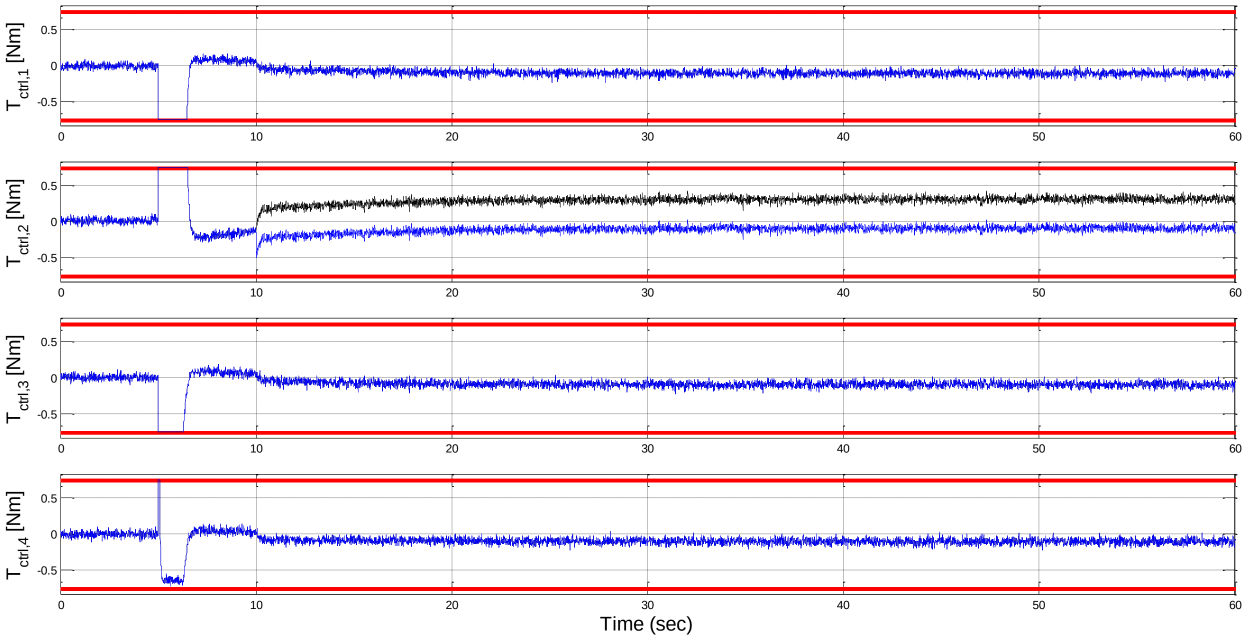


Figure 6.1 - Fault scenario n.1: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of step fault on the control input $T_{ctrl,2}$.

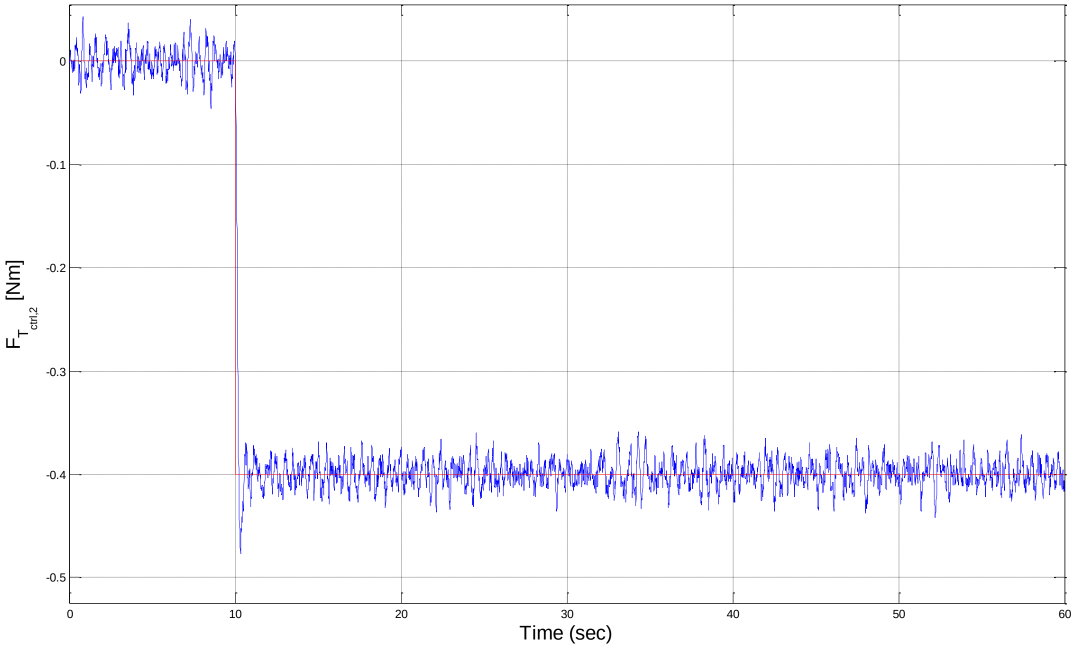


Figure 6.2 - Fault scenario n.1: true (red) and estimated (blue) step fault on the control input $T_{ctrl,2}$.

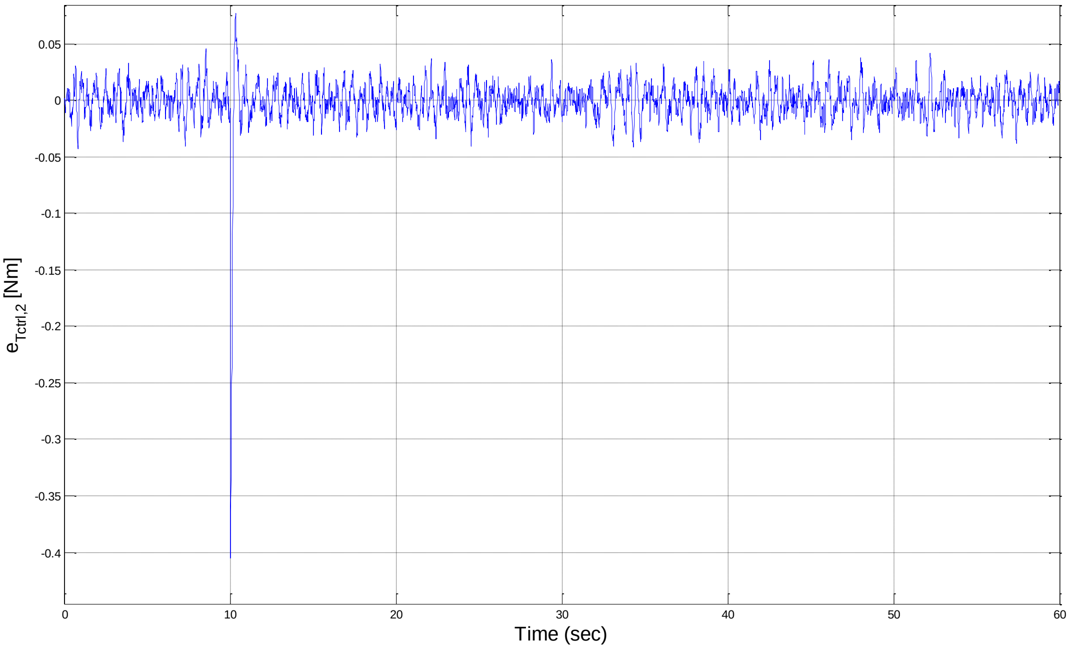


Figure 6.3 - Fault scenario n.1: estimation error for the step fault on the control input $T_{ctrl,2}$.

Figs. 6.4, 6.5 and 6.6 show the same results in case of a sinusoidal fault, whereas Figs. 6.7, 6.8 and 6.9 show the results in case of a rectangular pulse fault. Figs. 6.10, 6.11 and 6.12 show the obtained results in case of a ramp fault. In the step, rectangular pulse and ramp fault scenarios, the obtained estimations result to be highly accurate and characterized by small and bounded estimation errors mainly due to the measurement noise, with significant error peaks only in correspondence of abrupt variations of the fault functions to be estimated. In the sinusoidal fault scenario, the designed adaptive estimation filter results to be able to estimate quite accurately the actual sensor fault, with an estimation error characterized by a zero mean value and an oscillatory behaviour bounded between small values in comparison to the fault size, due to the time-varying fault signal to be estimated.

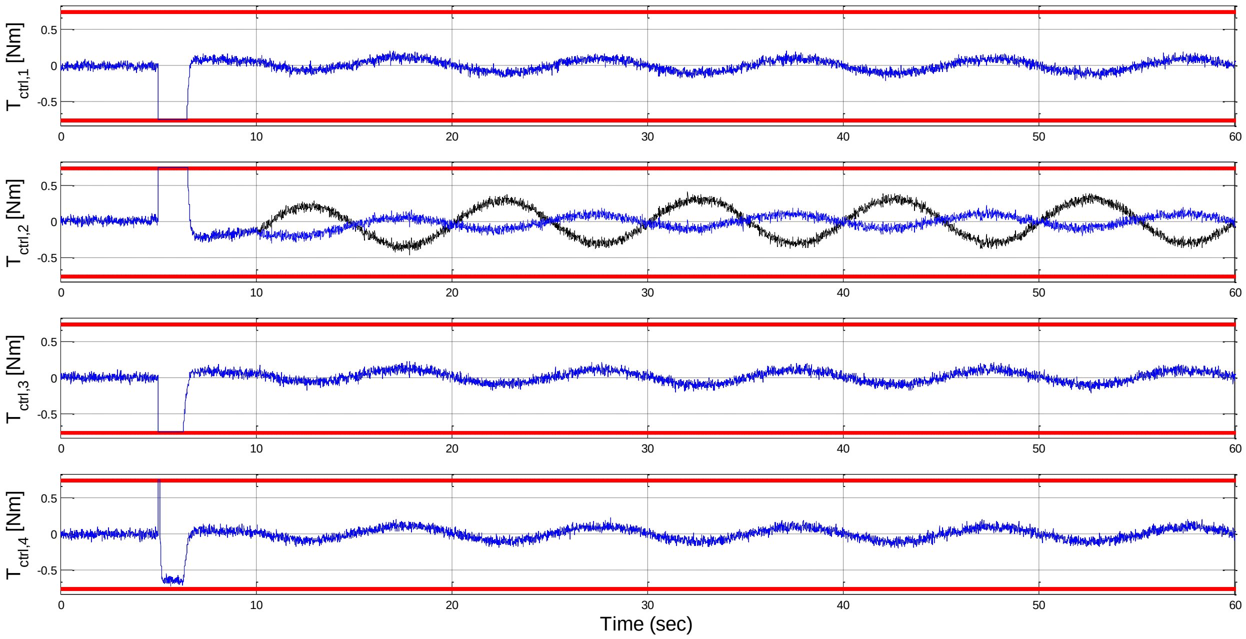


Figure 6.4 - Fault scenario n.2: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of sinusoidal fault on the control input $T_{ctrl,2}$.

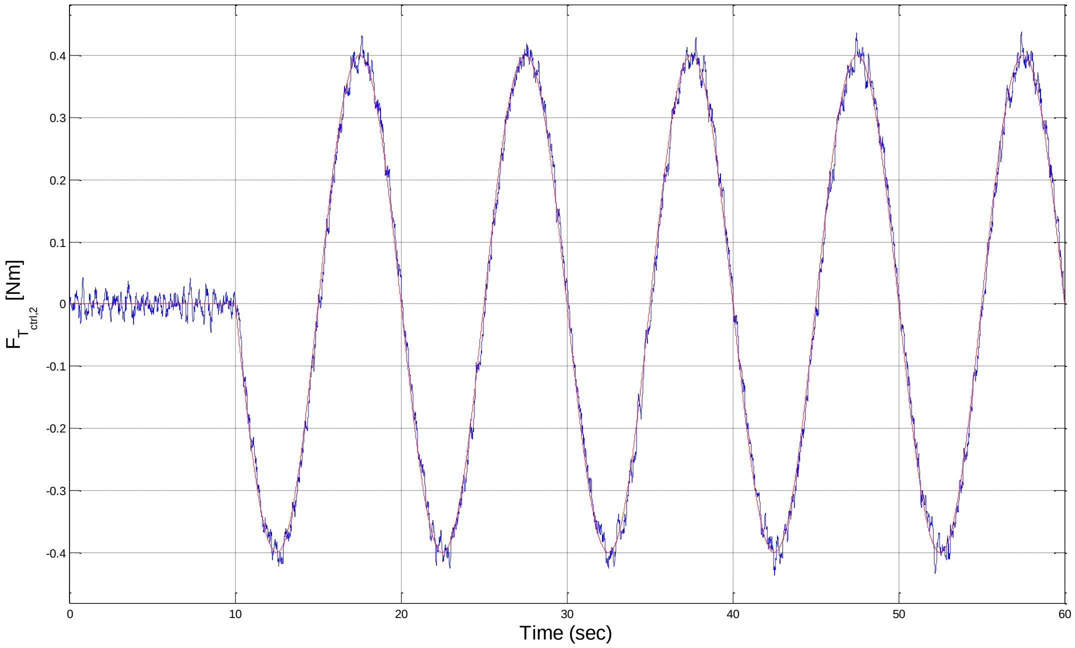


Figure 6.5 - Fault scenario n.2: true (red) and estimated (blue) sinusoidal fault on the control input $T_{ctrl,2}$.

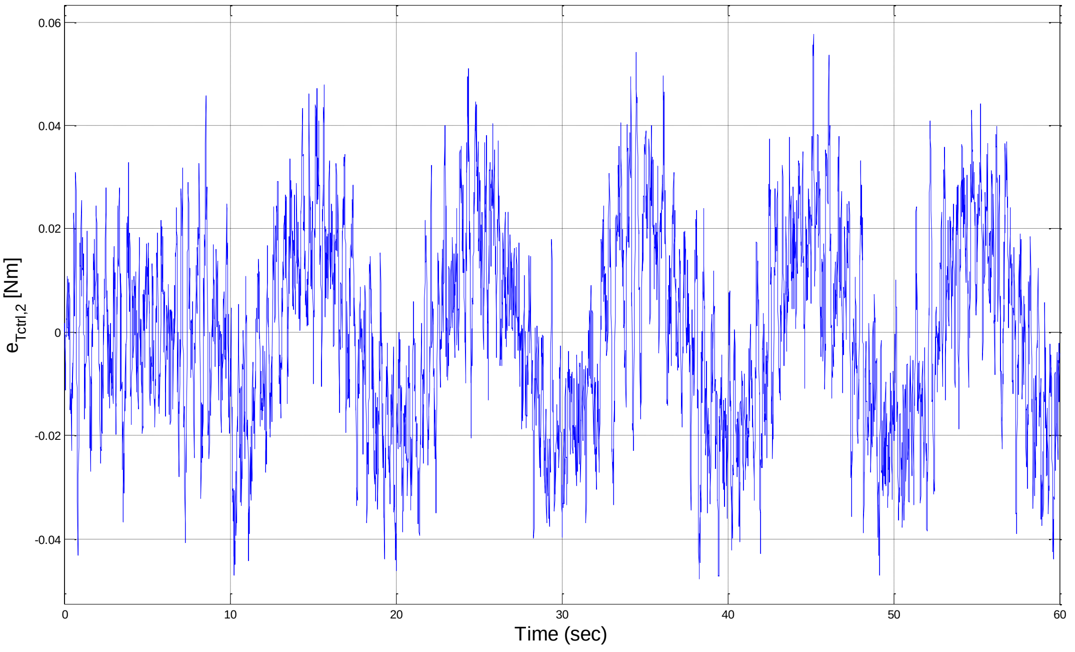


Figure 6.6 - Fault scenario n.2: estimation error for the sinusoidal fault on the control input $T_{ctrl,2}$.

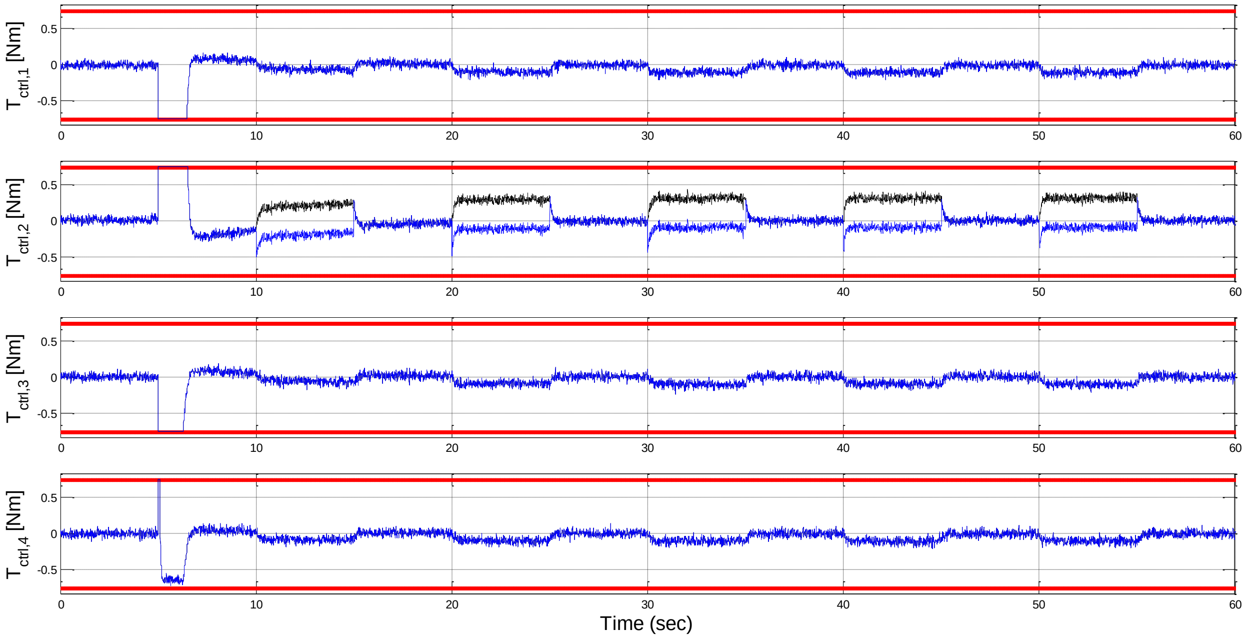


Figure 6.7 - Fault scenario n.3: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of rectangular pulse fault on the control input $T_{ctrl,2}$.

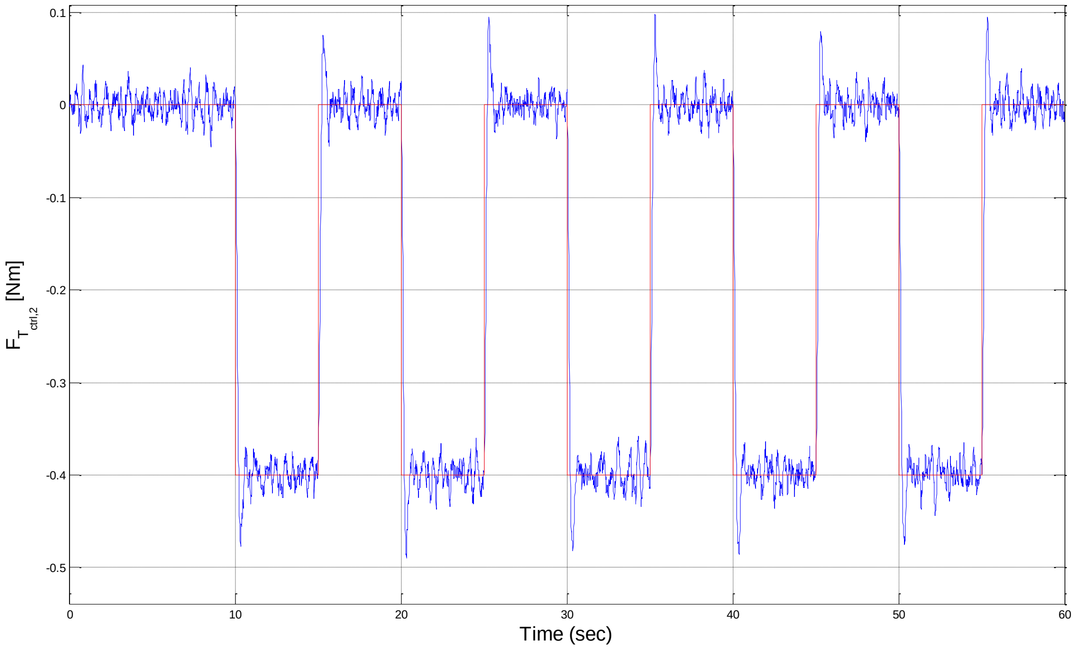


Figure 6.8 - Fault scenario n.3: true (red) and estimated (blue) rectangular pulse fault on the control input $T_{ctrl,2}$.

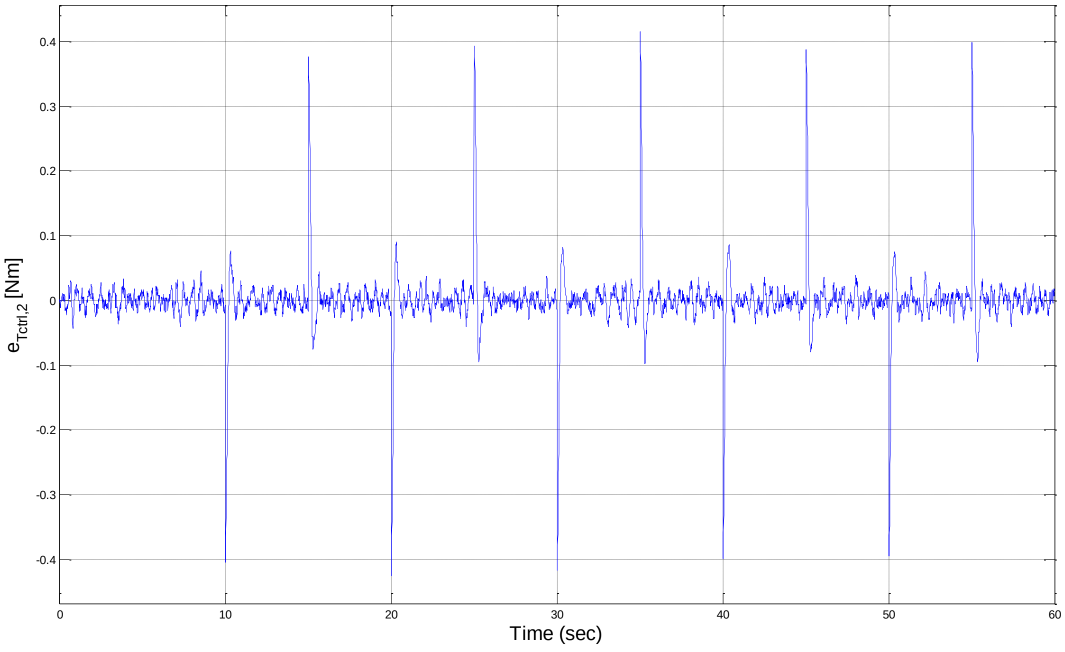


Figure 6.9 - Fault scenario n.3: estimation error for the rectangular pulse fault on the control input $T_{ctrl,2}$.

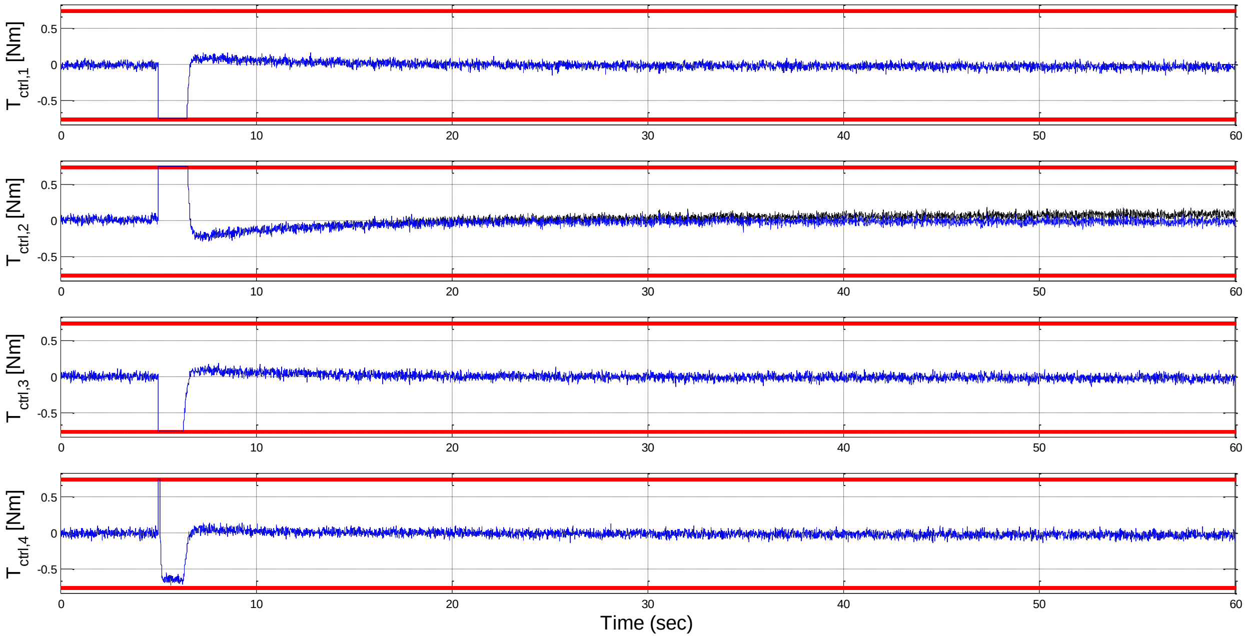


Figure 6.10 - Fault scenario n.4: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of ramp fault on the control input $T_{ctrl,2}$.

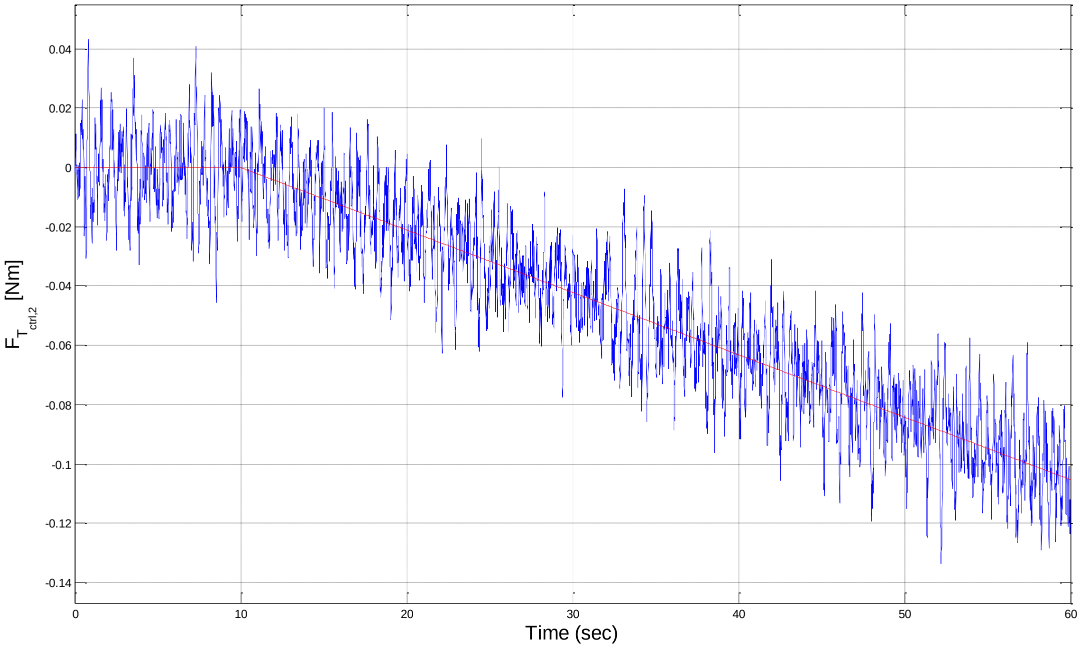


Figure 6.11 - Fault scenario n.4: true (red) and estimated (blue) ramp fault on the control input $T_{ctrl,2}$.

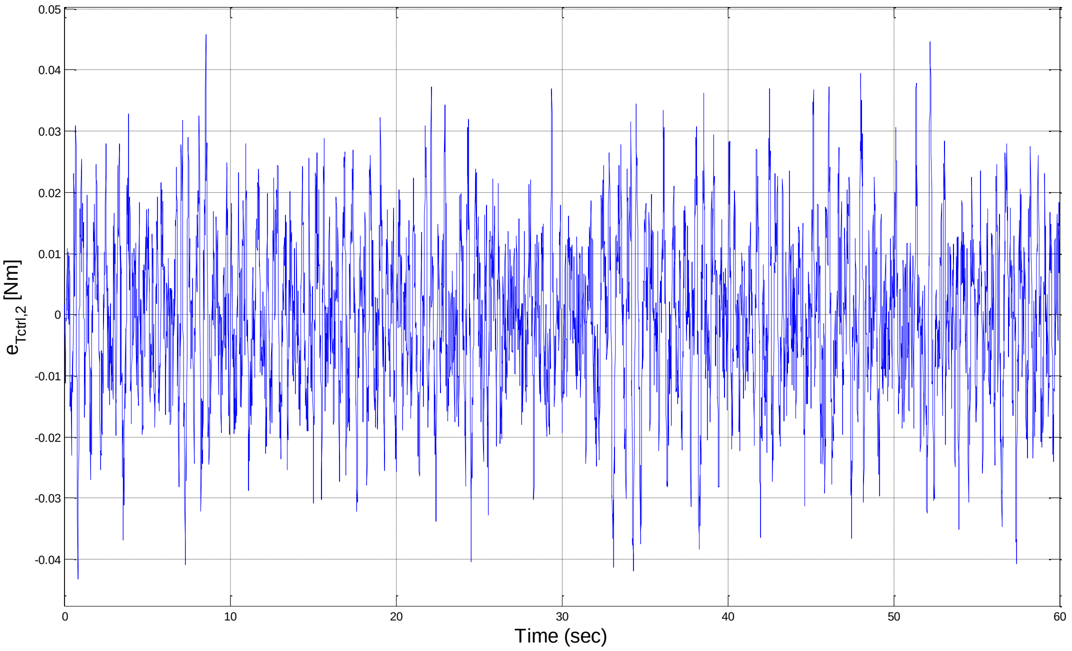


Figure 6.12 - Fault scenario n.4: estimation error for the ramp fault on the control input $T_{ctrl,2}$.

Finally, for the case of failure of the actuator providing the control torque $T_{ctrl,2}$, whose results are shown in Figs. 6.13, 6.14 and 6.15, the estimated fault represents the difference between the nominal control input commanded by the controller and the true actuated control torque, actually equal to zero.

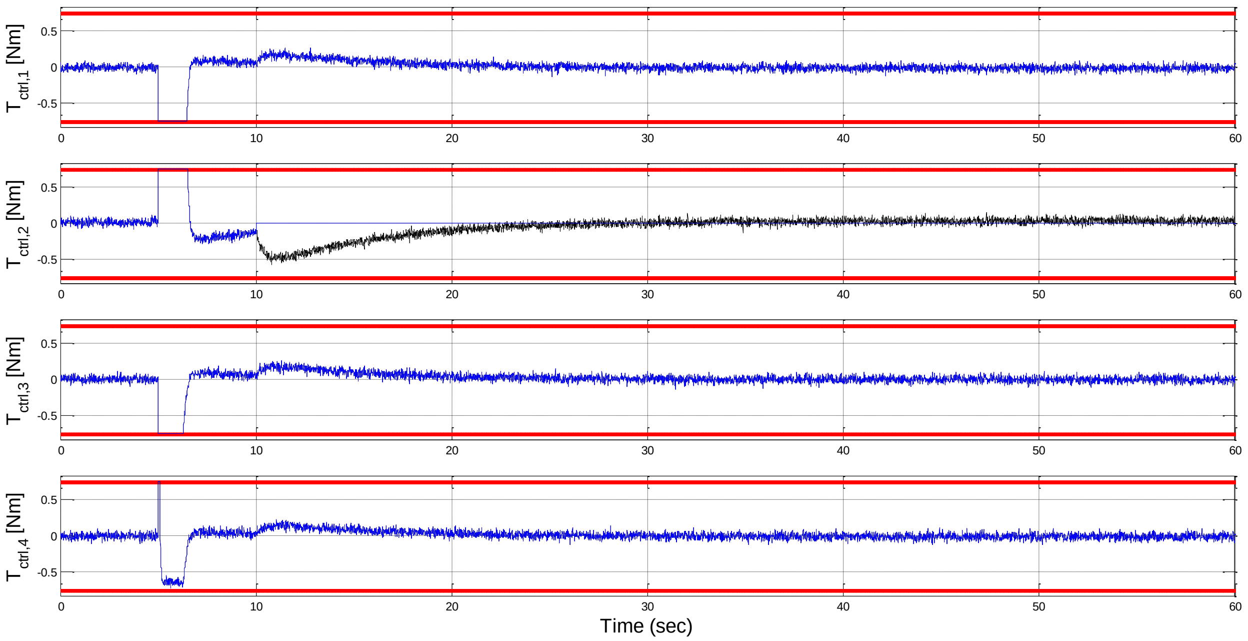


Figure 6.13 - Fault scenario n.5: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of failure of the actuator providing the control input $T_{ctrl,2}$.

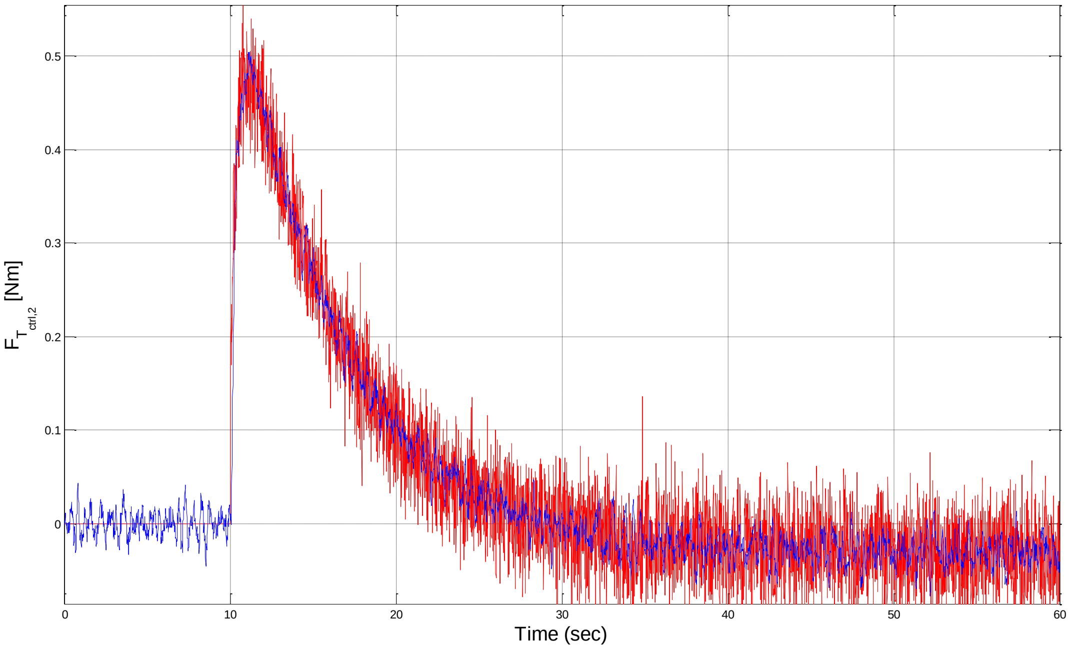


Figure 6.14 - Fault scenario n.5: true (red) and estimated (blue) fault corresponding to the failure of the actuator providing the control input $T_{ctrl,2}$.

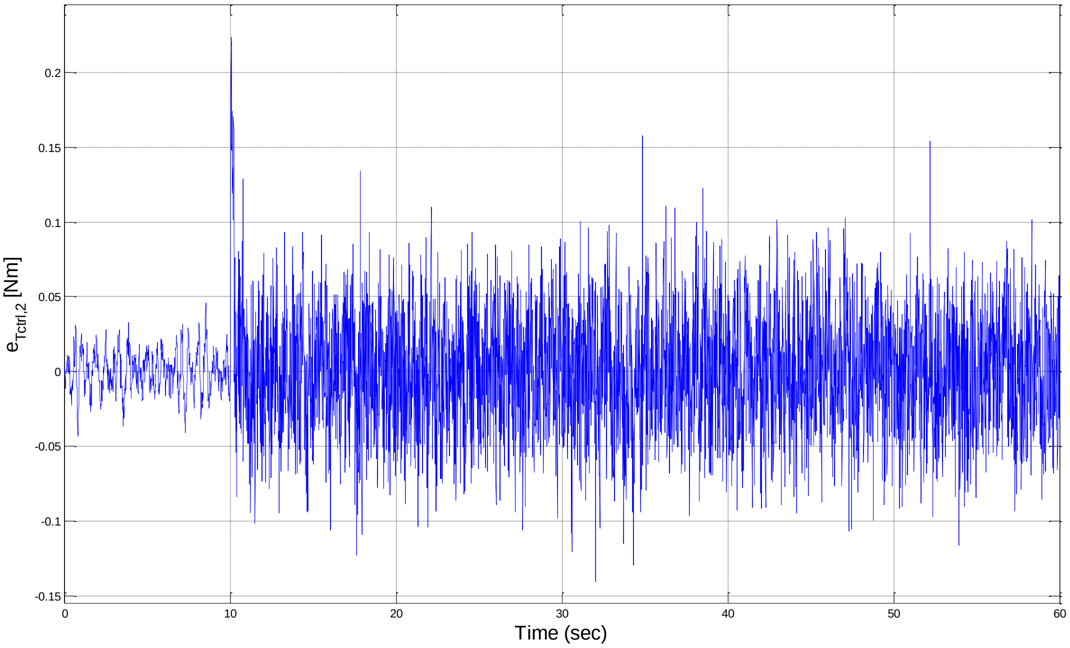


Figure 6.15 - Fault scenario n.5: estimation error of the fault corresponding to the failure of the actuator providing the control input $T_{ctrl,2}$.

Considering the occurrence of faults affecting the flywheel spin rate sensors, for the sixth fault scenario, *i.e.* the step fault affecting the measured spin rate $\omega_{w,3}$, Fig. 6.16 shows the effect of the fault on the sensor measurement, depicted in blue, with respect to the corresponding state variable depicted in black. The constant bias between the measured $\omega_{w,3_{measured}}$ and the true value $\omega_{w,3}$ is clearly visible after the fault occurrence. Fig. 6.17 shows the fault estimation obtained with the designed RBF-NN adaptive fault estimation filters and Fig. 6.18 shows the fault estimation error.

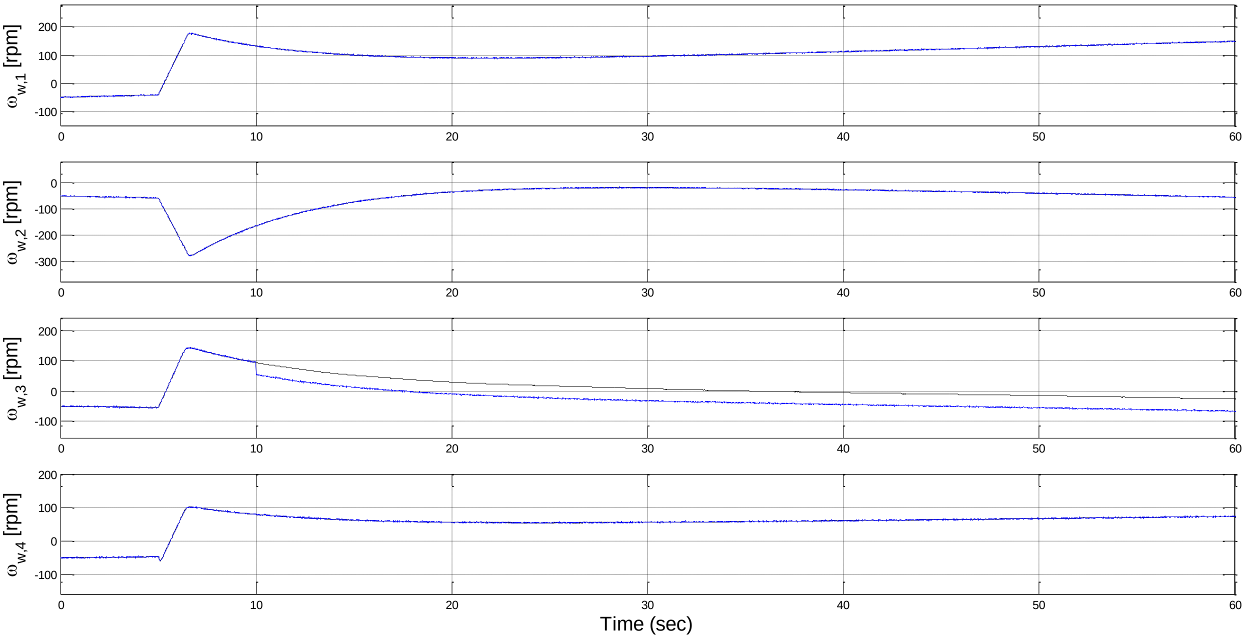


Figure 6.16 - Fault scenario n.6: true (black) and measured (blue) flywheel spin rates ω_w in case of step fault on the sensor output $\omega_{w,3_{measured}}$.

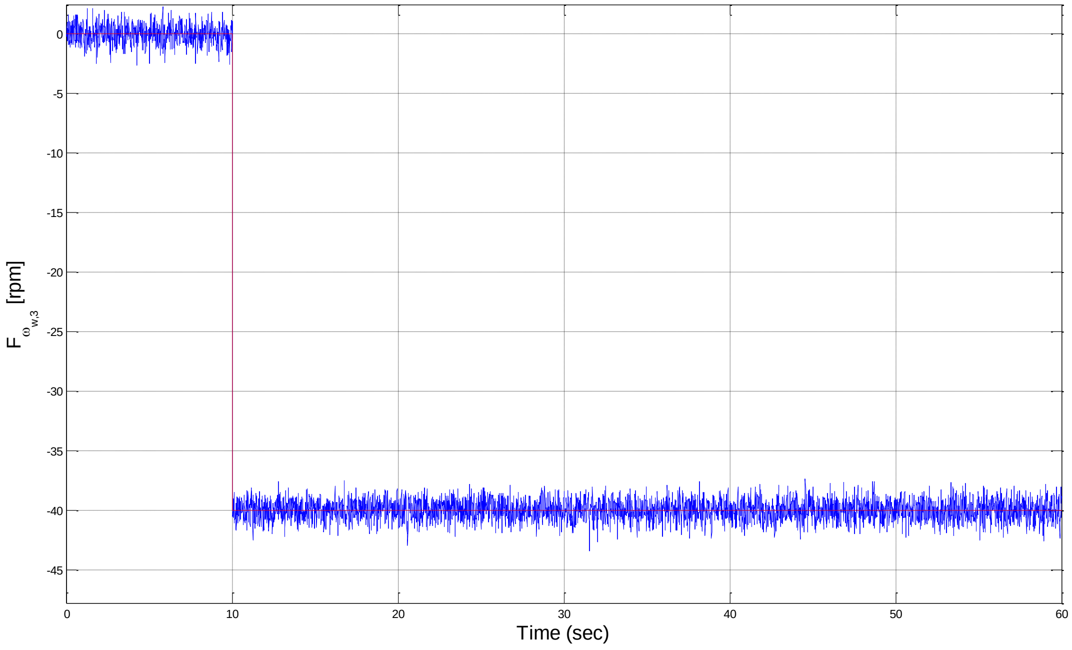


Figure 6.17 - Fault scenario n.6: true (red) and estimated (blue) step fault on the sensor output $\omega_{w,3_{measured}}$.

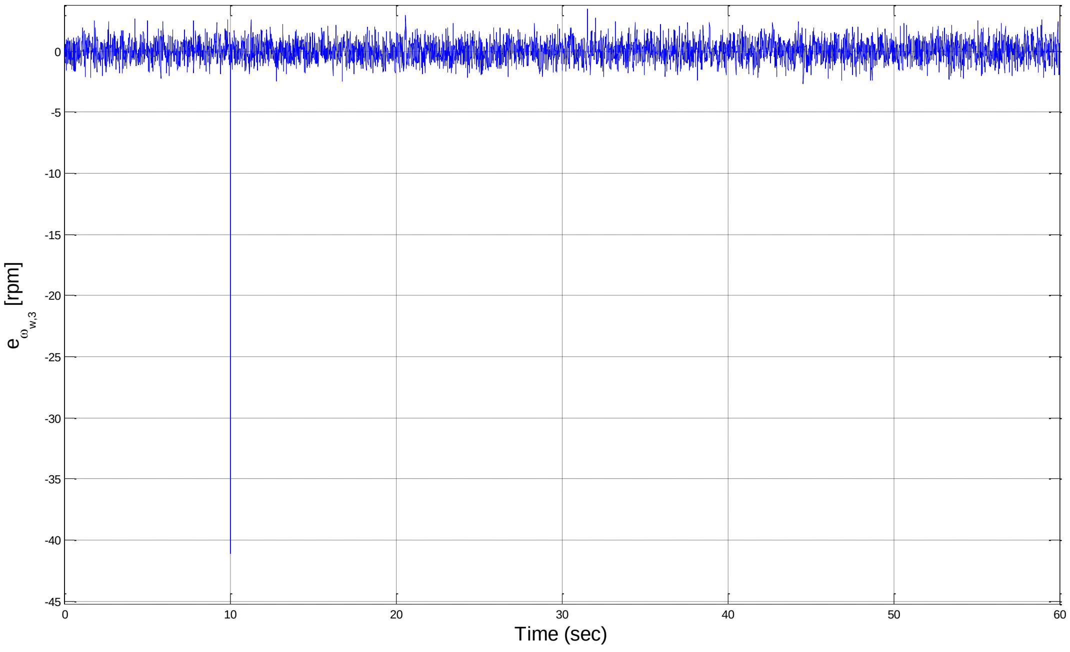


Figure 6.18 - Fault scenario n.6: estimation error for the step fault on the sensor output $\omega_{w,3_{measured}}$.

Figs. 6.19, 6.20 and 6.21 show the results in case of a sinusoidal sensor fault. The designed adaptive estimation filter results to be able to estimate quite accurately the actual sensor fault, with an estimation error characterized by a zero mean value and an oscillatory behaviour bounded between small values in comparison to the fault size, due to the time-varying fault signal to be estimated.

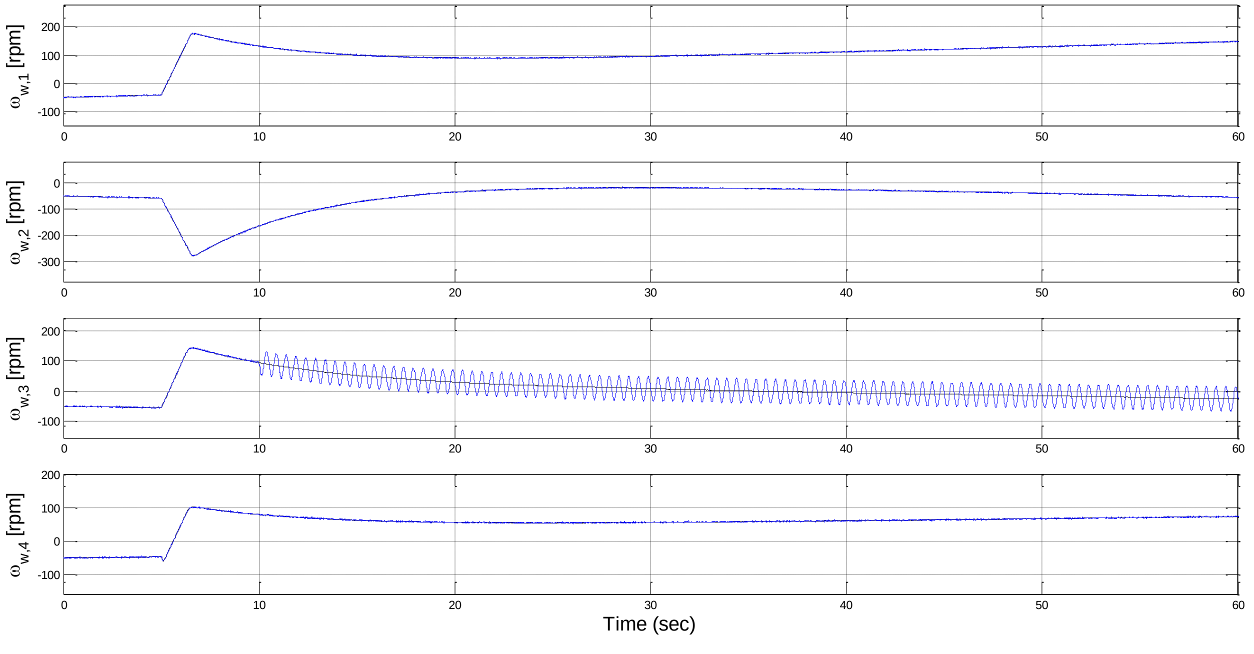


Figure 6.19 - Fault scenario n.7: true (black) and measured (blue) flywheel spin rates ω_w in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$.

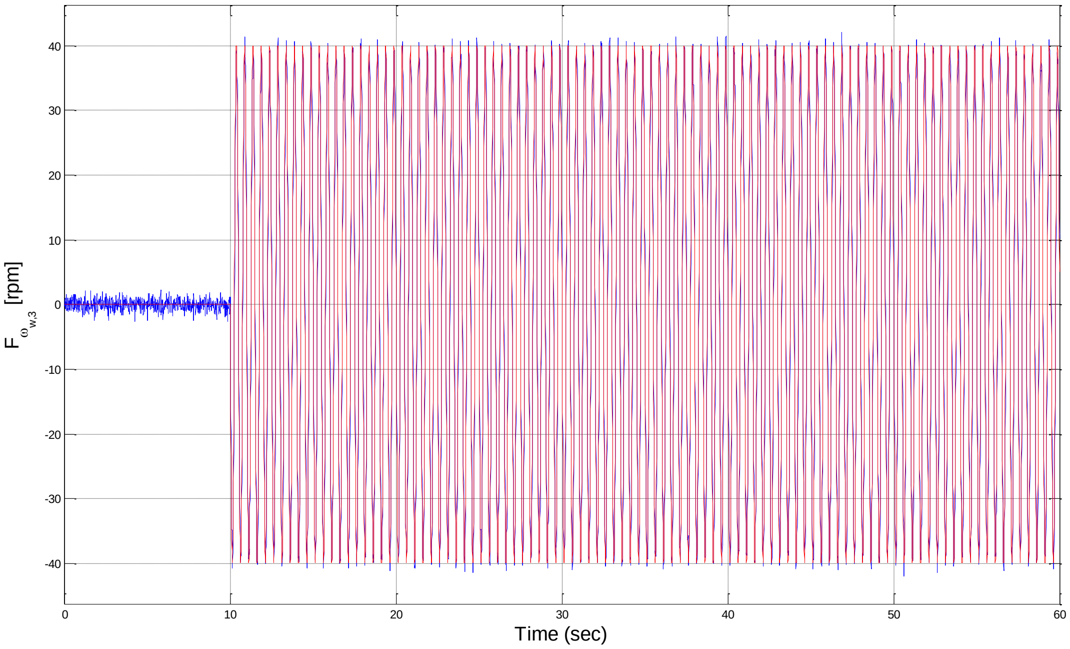


Figure 6.20 - Fault scenario n.7: true (red) and estimated (blue) sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$.

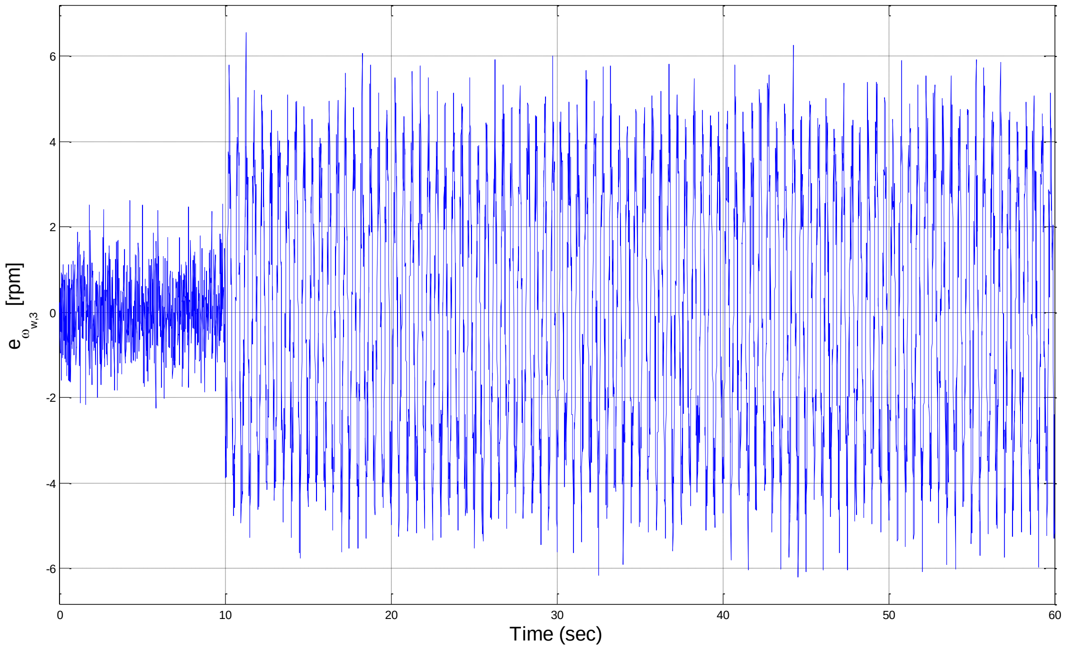


Figure 6.21 - Fault scenario n.7: estimation error for the sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$.

For the eighth fault scenario, whose results are shown in Figs. 6.22, 6.23 and 6.24, the estimated fault represents the difference between the true value of the state variable and the sensor measurement, actually equal to zero. Hence, the estimated fault results to correspond to the actual value of the state variable to be measured. Therefore, this estimation can be considered as the output of a software sensor.

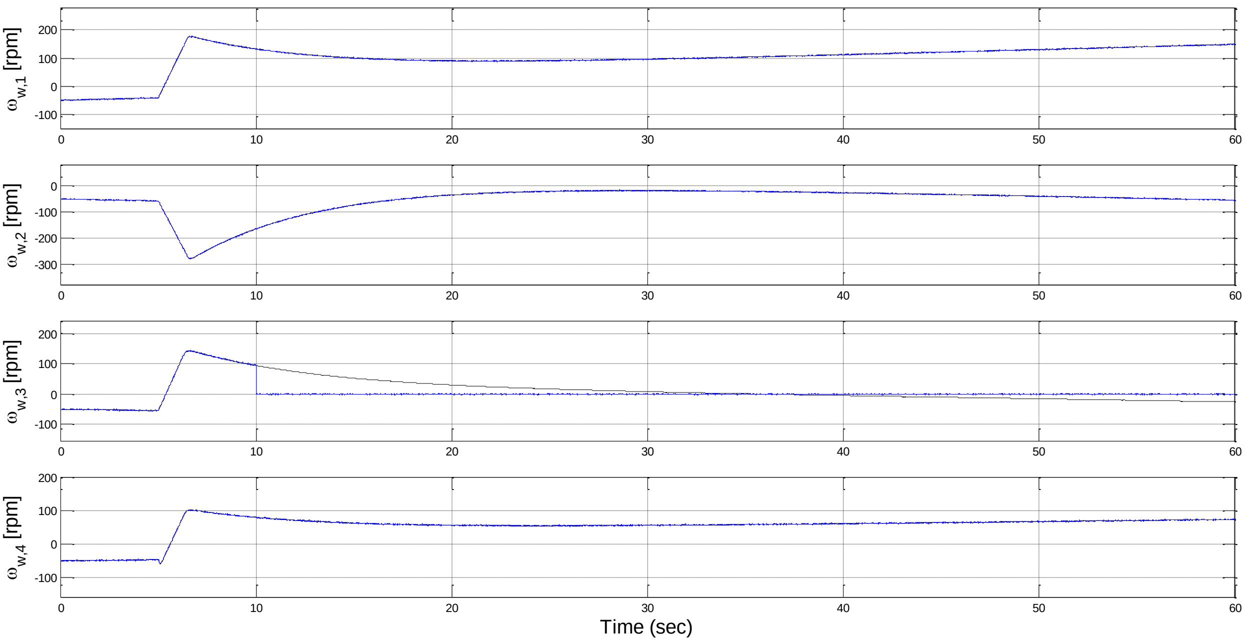


Figure 6.22 - Fault scenario n.8: true (black) and measured (blue) flywheel spin rates ω_w in case of failure of the sensor measuring $\omega_{w,3}$.

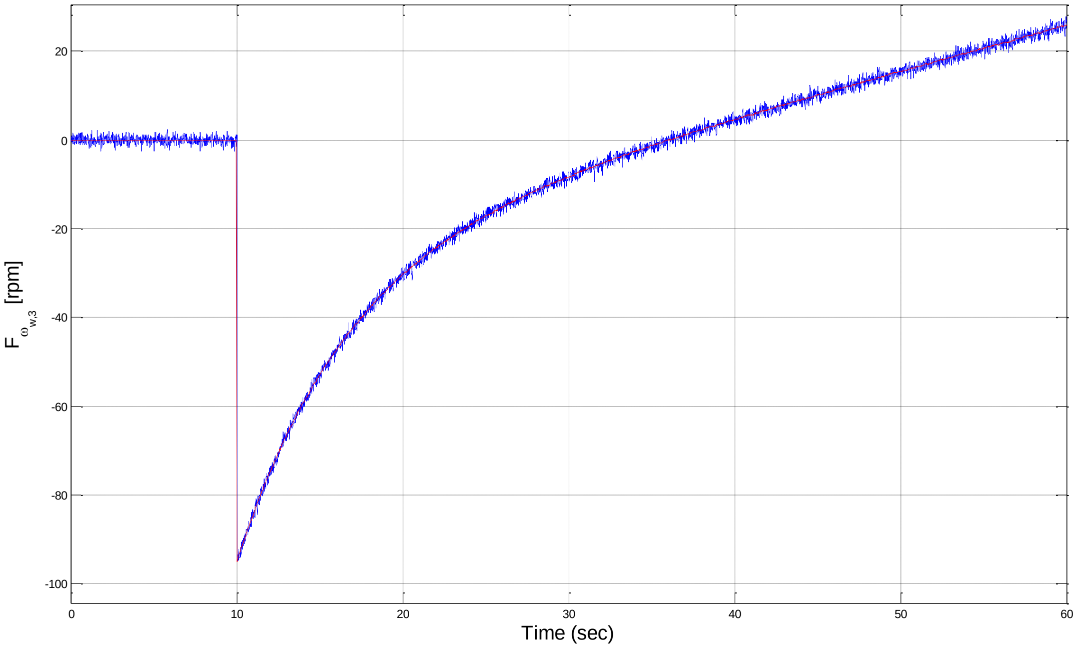


Figure 6.23 - Fault scenario n.8: true (red) and estimated (blue) fault corresponding to the failure of the sensor measuring $\omega_{w,3}$.

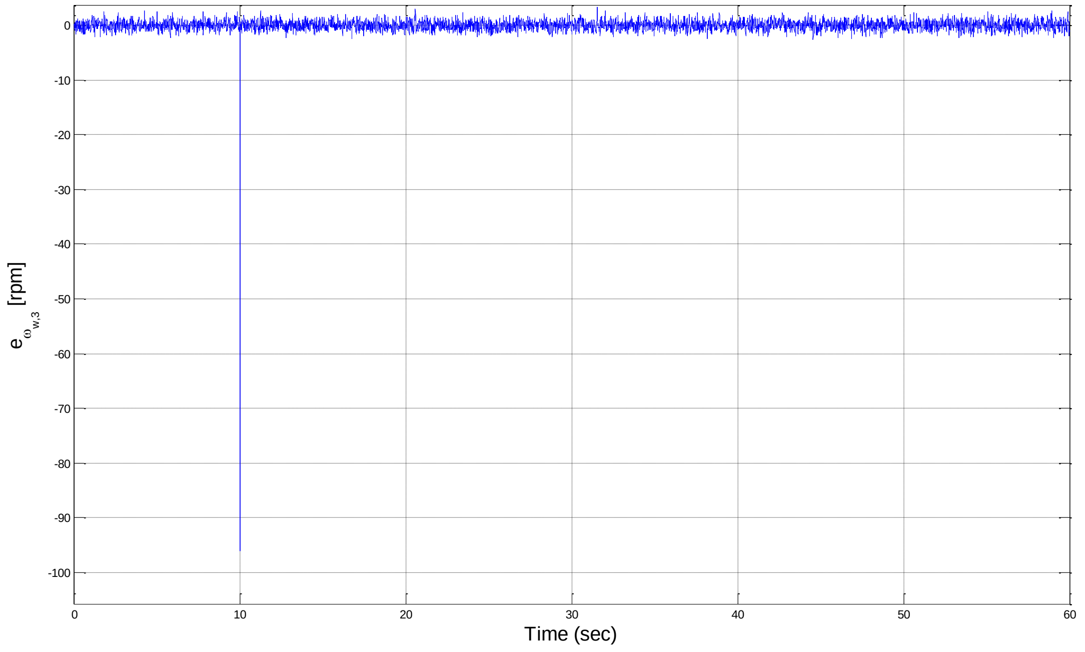


Figure 6.24 - Fault scenario n.8: estimation error of the fault corresponding to the failure of the sensor measuring $\omega_{w,3_{measured}}$.

In all these three fault scenarios, the obtained estimations result to be highly accurate and characterized by small and bounded estimation errors due to the measurement noise, with significant error peaks only in correspondence of abrupt variations of the fault functions to be estimated.

6.3.2 Estimation of Faults in the Satellite ADS

Now, considering the occurrence of faults affecting the sensors of the satellite ADS, for the ninth fault scenario, *i.e.* the step fault affecting the measured angular velocity $\omega_{x,i}$, Fig. 6.25 shows the effect of the fault on the sensor measurement, depicted in blue, with respect to the corresponding state variable depicted in black. The constant bias between the measured $\omega_{x,i_{measured}}$ and the true value $\omega_{x,i}$ is clearly visible after the fault occurrence. Fig. 6.26 shows the fault estimation obtained by exploiting the designed adaptive fault estimation filters based on the RBF-NN. Moreover, the obtained fault estimation error is shown in Fig. 6.27.

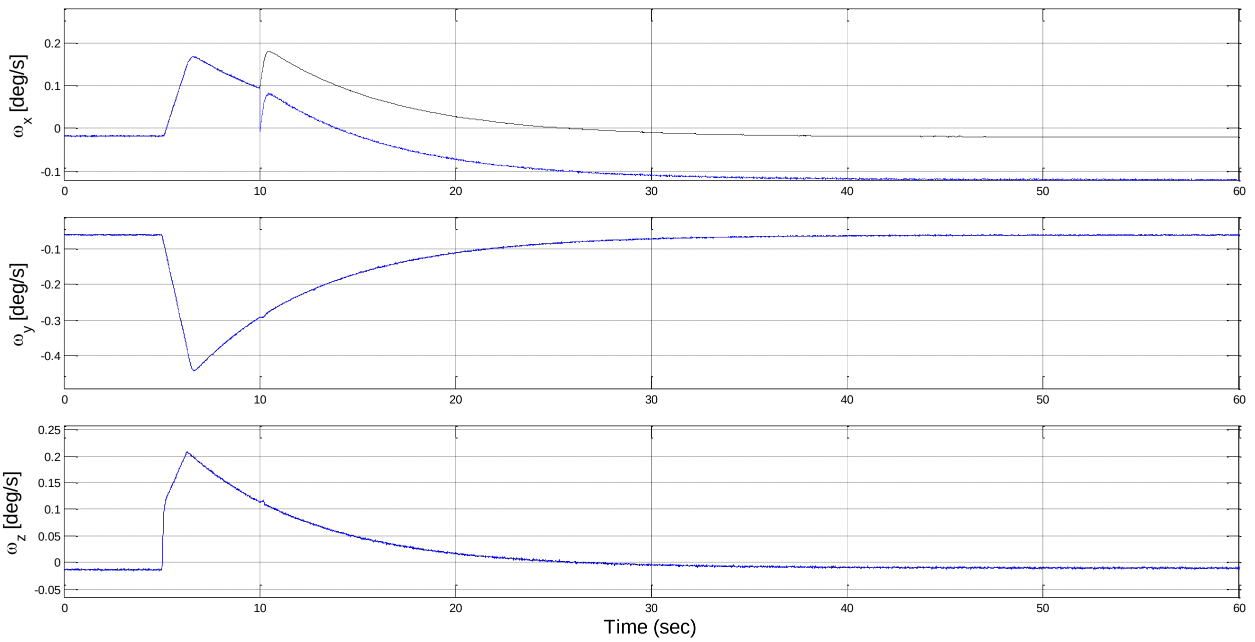


Figure 6.25 - Fault scenario n.9: true (black) and measured (blue) satellite angular velocity ω in case of step fault on the sensor output $\omega_{x,i_{measured}}$.

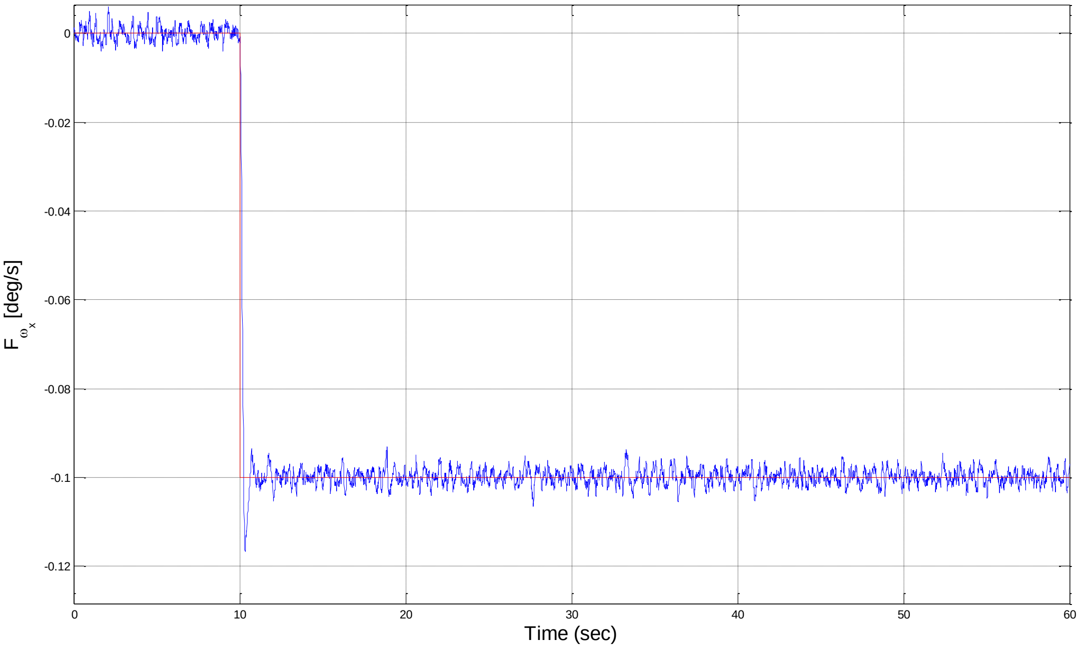


Figure 6.26 - Fault scenario n.9: true (red) and estimated (blue) step fault on the sensor output $\omega_{x,i_{measured}}$.

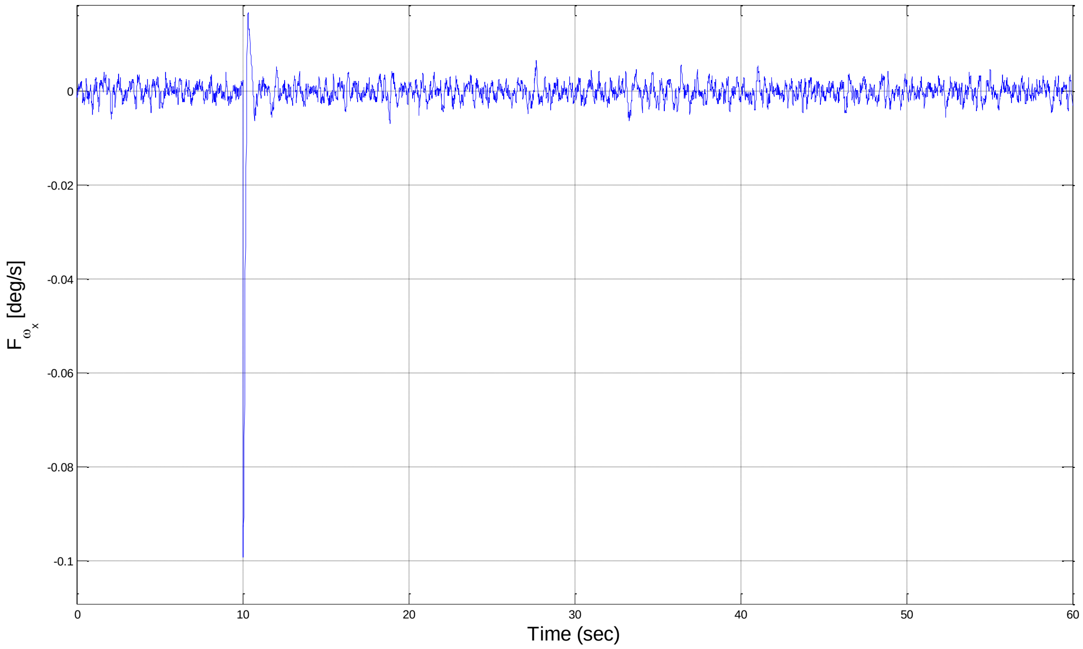


Figure 6.27 - Fault scenario n.9: estimation error for the step fault on the sensor output $\omega_{x,i_{measured}}$.

Figs. 6.28, 6.29, 6.30, 6.31, 6.32 and 6.33 show the estimation results in case the tenth and eleventh sensor fault scenarios, *i.e.* a lock-in-place (stuck) of the sensor measurement $\omega_{x,i}$ and sensor loss of effectiveness, respectively. In the lock-in-place fault scenario, the designed adaptive estimation filter results to be able to estimate quite accurately the actual sensor fault, with an estimation error characterized by a zero mean value and an oscillatory behaviour bounded between small values in comparison to the fault size, due to the time-varying fault signal to be estimated.

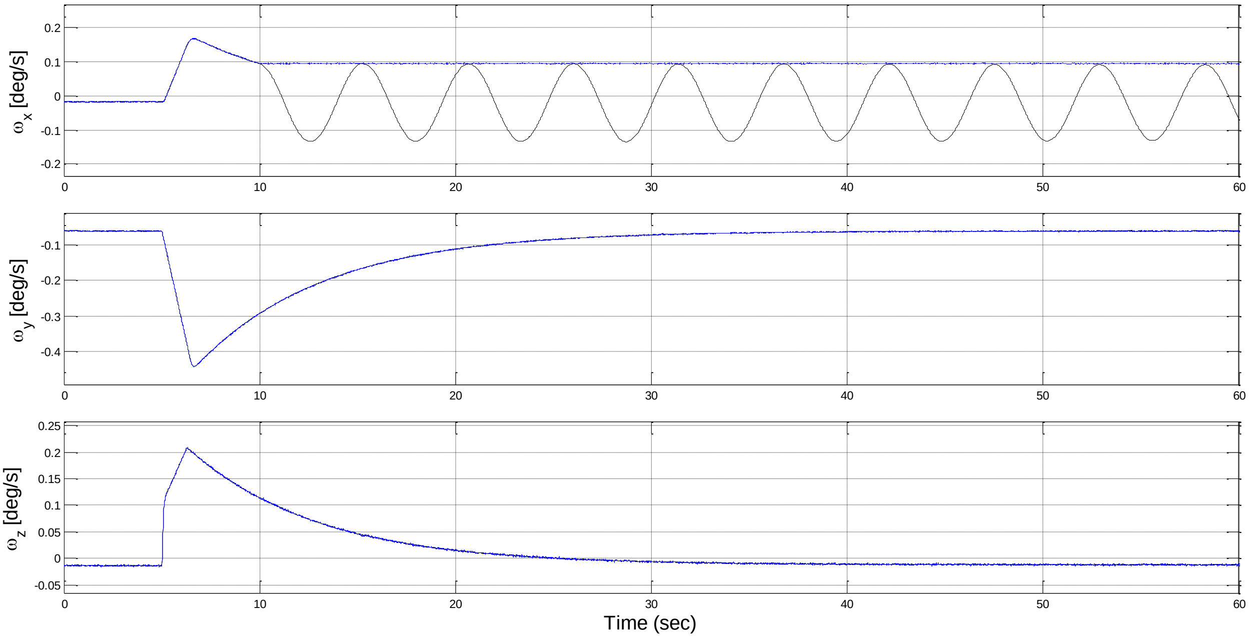


Figure 6.28 - Fault scenario n.10: true (black) and measured (blue) satellite angular velocity ω in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

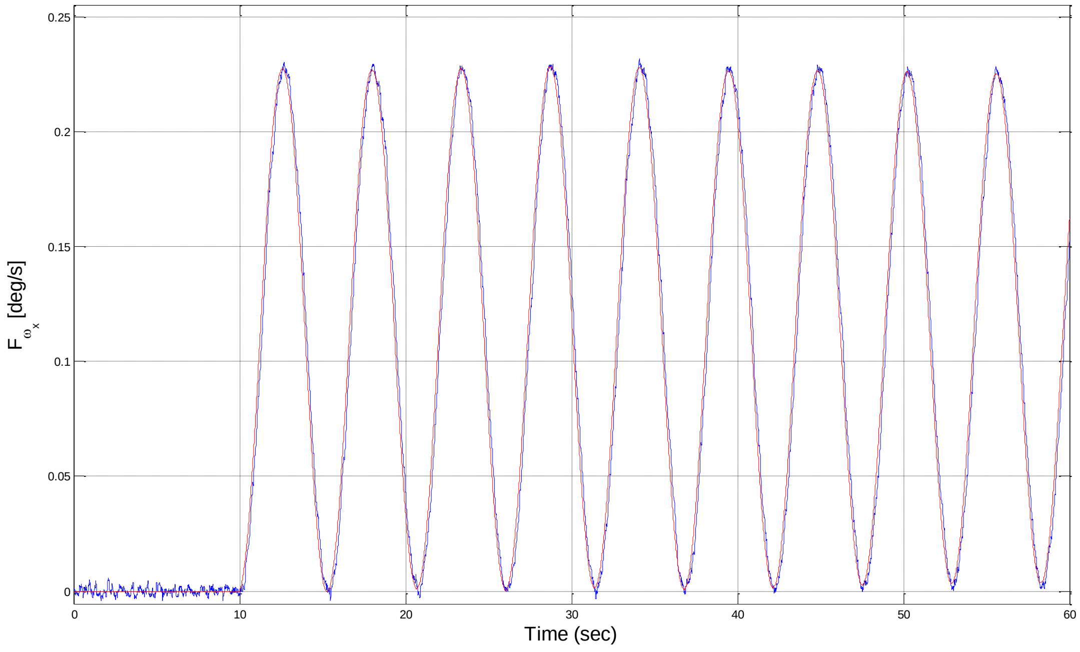


Figure 6.29 - Fault scenario n.10: true (red) and estimated (blue) lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

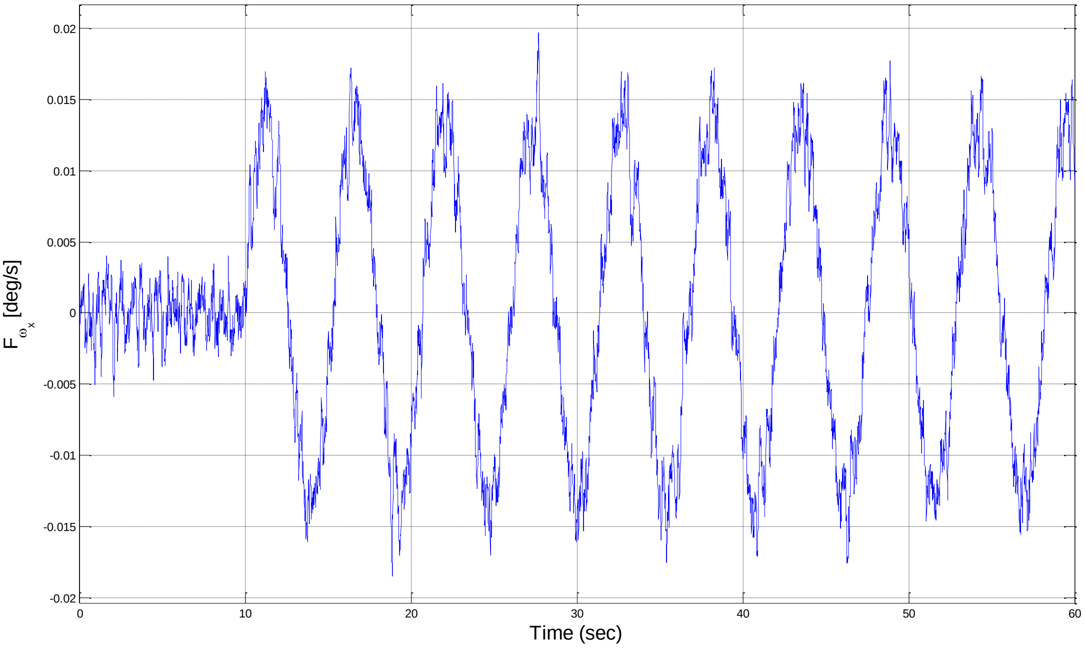


Figure 6.30 - Fault scenario n.10: estimation error for the lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$.

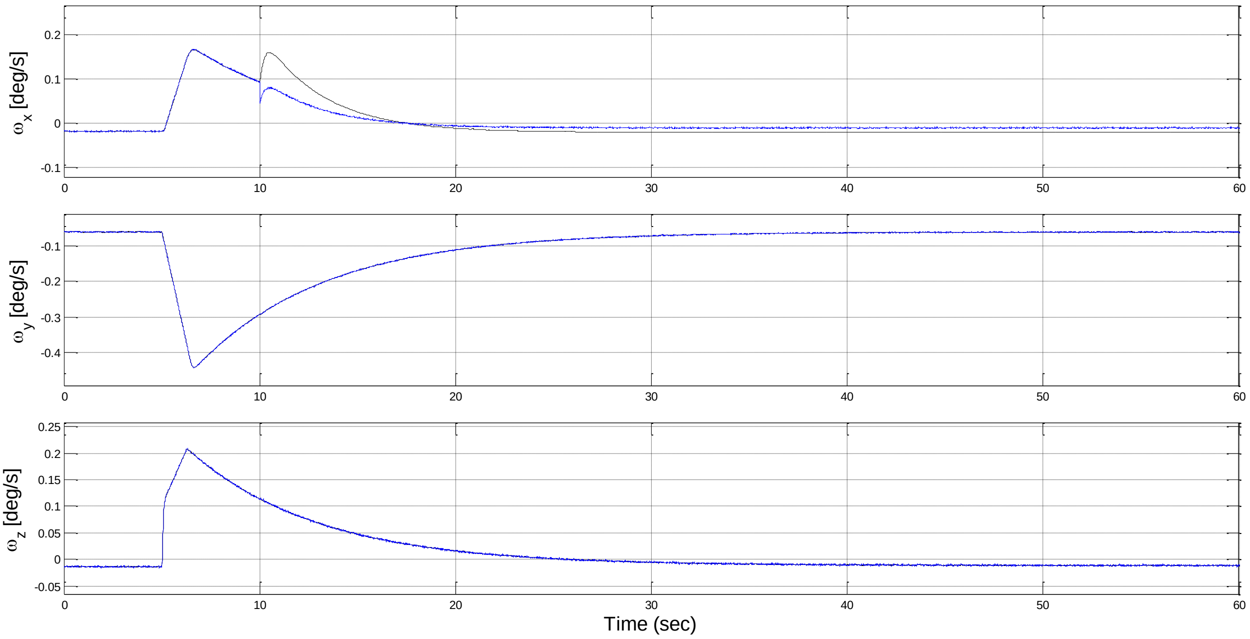


Figure 6.31 - Fault scenario n.11: true (black) and measured (blue) satellite angular velocity ω in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

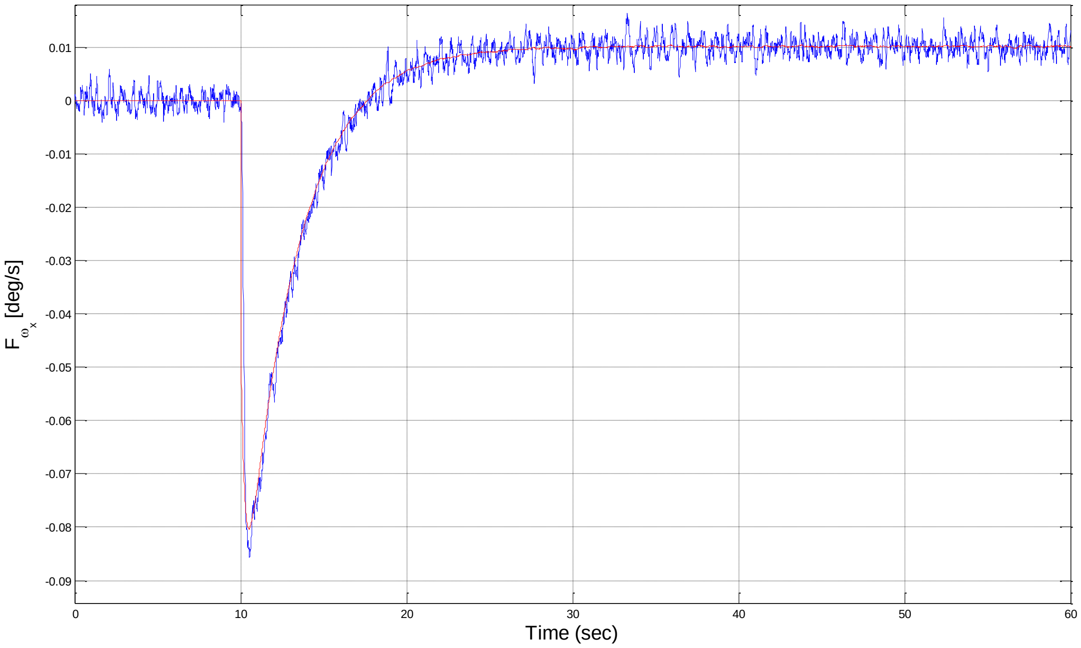


Figure 6.32 - Fault scenario n.11: true (red) and estimated (blue) loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

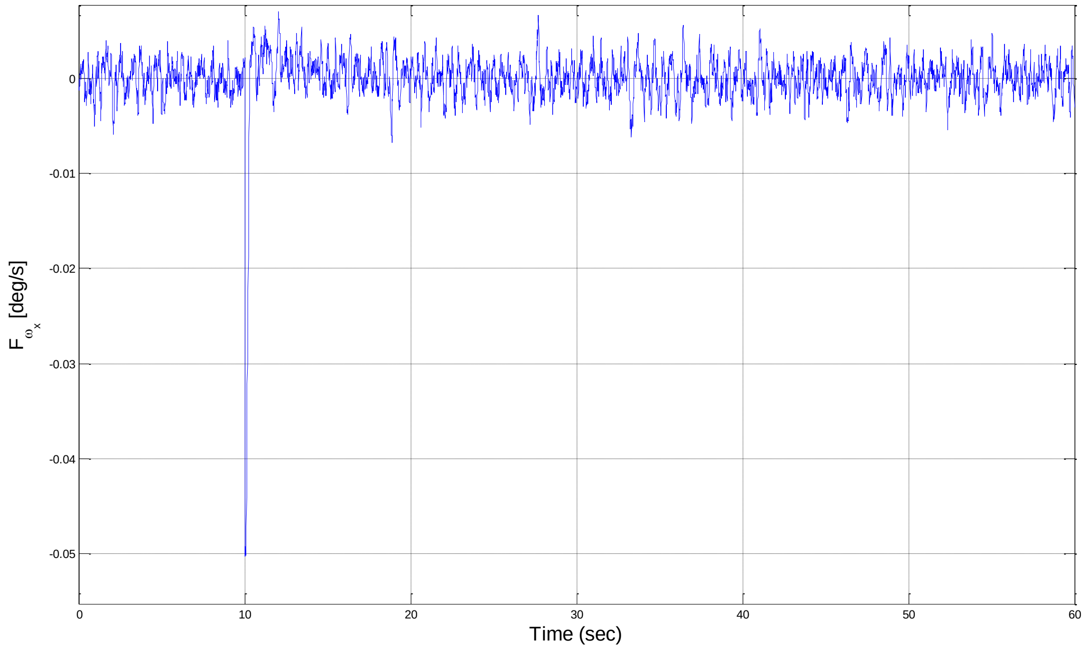


Figure 6.33 - Fault scenario n.11: estimation error for the loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$.

Finally, considering the twelfth fault scenario, *i.e.* the additive fault $\bar{\mathbf{F}}_{\bar{\mathbf{q}}_{star1}}$ on the sensor measurement of $\bar{\mathbf{q}}_{star1}$, Figs. 6.34, 6.35 and 6.36 show the obtained estimation results and the estimation error of the additive quaternion fault vector $\bar{\mathbf{F}}_{\bar{\mathbf{q}}_{star1}}$.

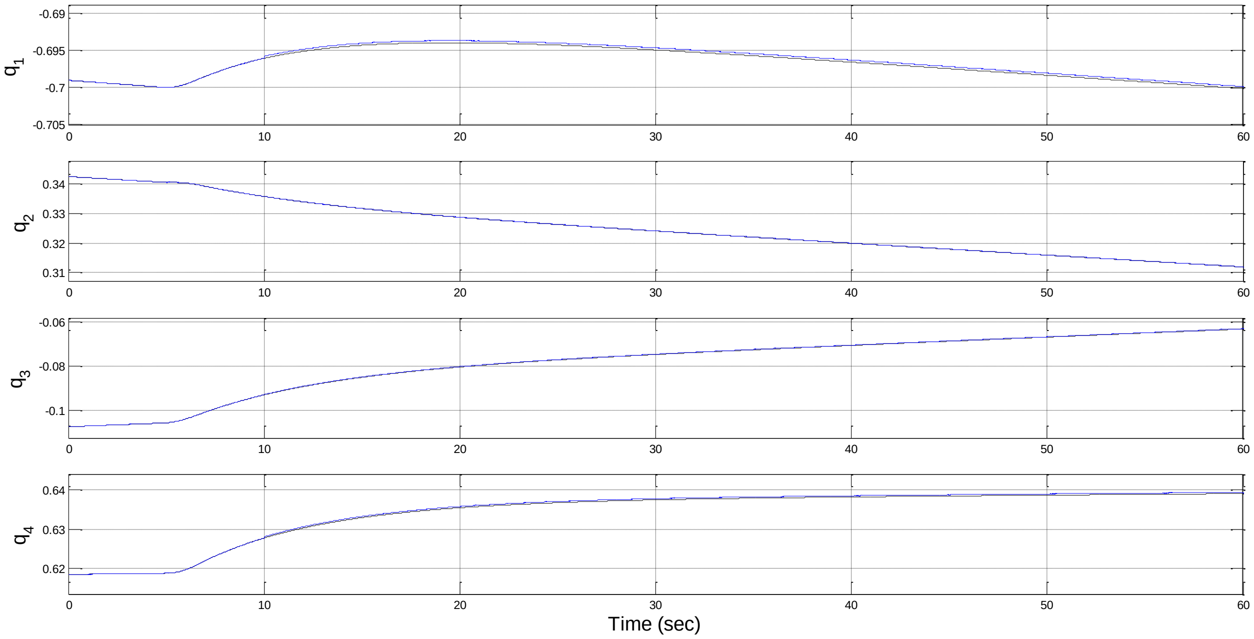


Figure 6.34 - Fault scenario n.12: true (black) and measured (blue) attitude quaternion vector $\bar{\mathbf{q}}_{star1}$ in case of fault on the first attitude sensor output.

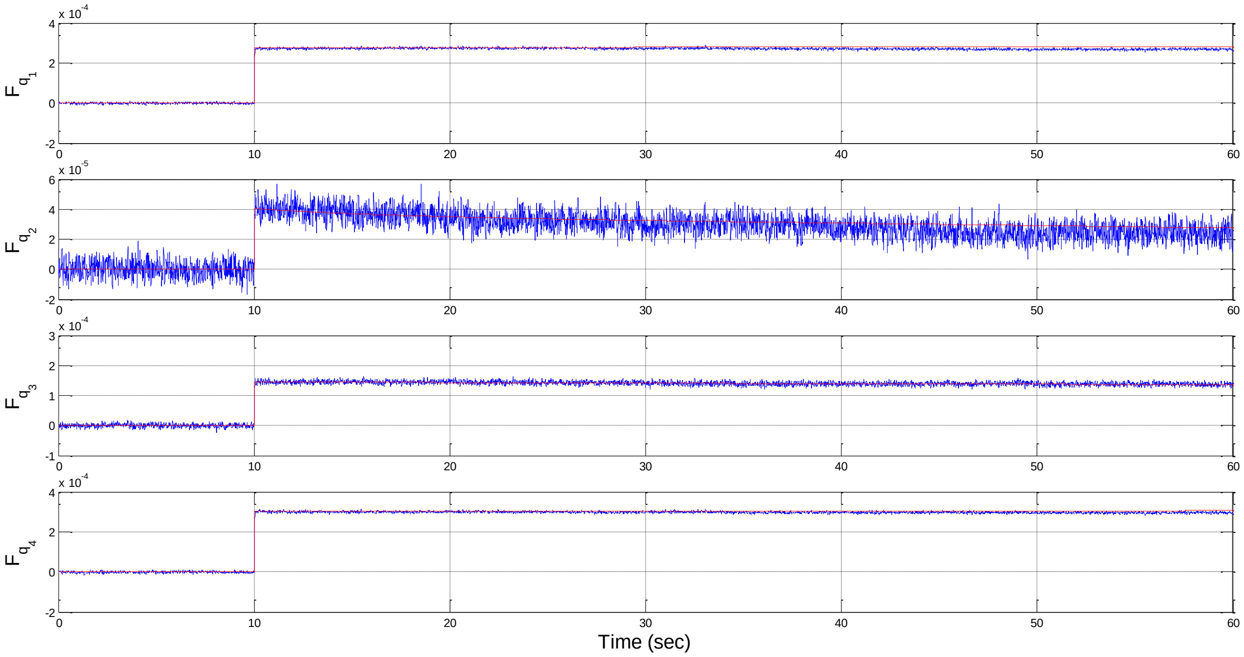


Figure 6.35 - Fault scenario n.12: true (red) and estimated (blue) additive fault vector associated to the physical fault on the sensor measuring the attitude quaternion vector $\bar{\mathbf{q}}_{star1}$.

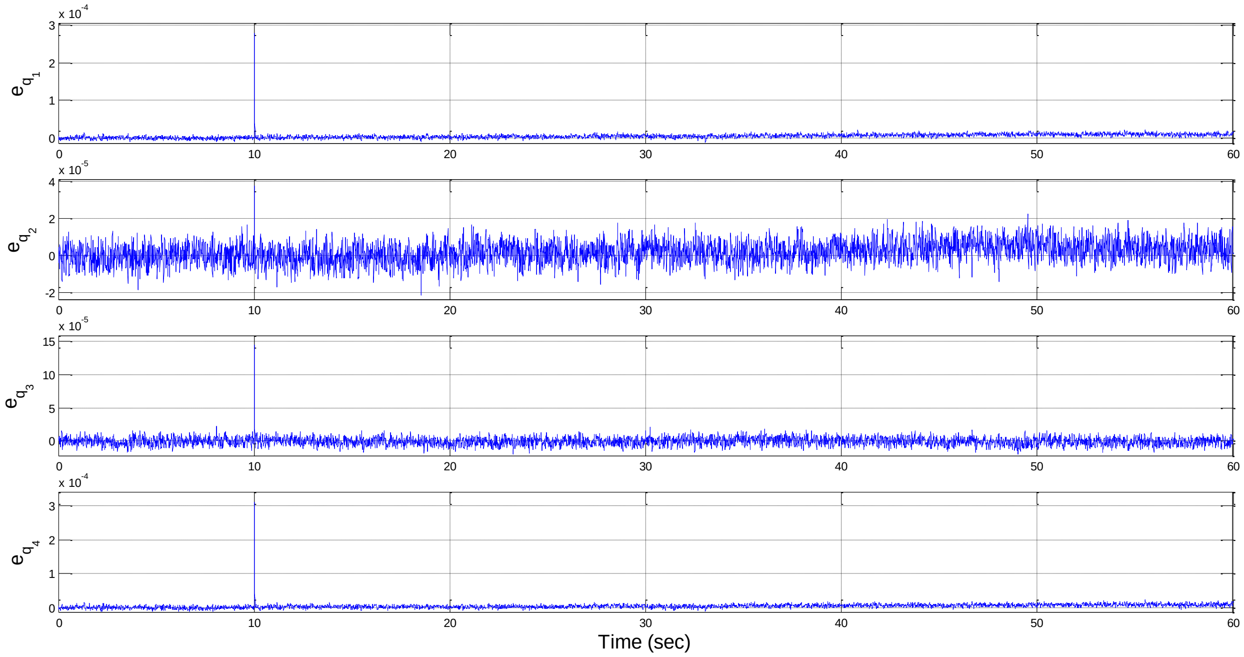


Figure 6.36 - Fault scenario n.12: estimation error of the additive fault vector associated to the physical fault on the sensor measuring the attitude quaternion vector $\bar{\mathbf{q}}_{star1}$.

7 FAULT TOLERANT CONTROL

This chapter provides a general overview of the FTC problem. A classification of the main approaches is presented, in particular focusing on the active fault tolerant methods, along with the definition of the main regions of operation. The general description of the active fault tolerant control methods exploited in this thesis is also provided.

7.1 General Overview

Modern complex systems rely on sophisticated control systems to meet increased performance and safety requirements. A conventional feedback control design for a complex system may result in an unsatisfactory performance, or even instability, in the event of malfunctions in system components.

Actuators and sensors, for example, may be affected by partial or complete loss of effectiveness faults, as well as offsets and stuck-at faults during operation. These faults may degrade the system performance with respect to the nominal fault-free one or, in the worst case, end up with loss of stability. To overcome such weaknesses, new approaches to control system design have been developed in order to tolerate component malfunctions while maintaining desirable stability and performance properties in the presence of faults (Blanke *et al.* 2006; Noura *et al.* 2009; Edwards *et al.* 2010; Richter 2011; Zhang *et al.* 2009; Witczak 2014).

This is particularly important for safety-critical systems, such as aircrafts, spacecrafts, nuclear power plants, and chemical plants processing hazardous materials. In such systems, the consequences of a minor fault in a system component can be catastrophic. Therefore, the demand on reliability, safety and fault tolerance is generally high. It is necessary to design control systems which are capable of tolerating potential faults in these systems in order to improve the reliability, availability, maintainability and survivability of the controlled system while providing a desirable performance. These types of control systems are often known as *Fault Tolerant Control Systems* (FTCS). More precisely, FTCS are control systems that possess the ability to accommodate component faults automatically. They are capable of maintaining overall system stability and acceptable performance (*i.e.* an acceptable degradation of system performance) in the event of such faults. In other words, a closed-loop control system, which can tolerate component malfunctions, while maintaining desirable performance and stability properties, is said to be a fault tolerant control system.

The accommodation capability of a control system depends on many factors such as the severity of the failure, the robustness of the nominal system and the presence of mechanisms that introduce redundancy in actuators and/or sensors. Concerning the type of control reconfiguration to be adopted, it strongly depends on the fault severity. It can consider the complete shut-off of the faulty component if the occurred fault is severe, or it could limit to small reconfigurations of the control structure or controller in case of soft faults, aiming to preserve the required performances.

Even though a temporary degradation of the performance of the overall system is acceptable, the primary objective of FTC consists in maintaining the system operative for a sufficiently long interval such that a human operator or the automatic system itself are able to intervene and repair the damaged system or perform the necessary actions in order to avoid dangerous and more severe, or even irremediable, consequences. In Fig. 7.1, the following three main regions of system operation are shown:

Region of required performance: the region in which the system satisfy the operation (mission objectives). The nominal controller maintains the system in this region, despite of the presence of disturbances and model uncertainties. Moreover, the system could remain in this region even in case of small faults;

Region of degraded performance: the region in which the fault takes the system to operate. This region denotes the zone where the faulty system can remain without excessive risks. The FTC must be able to initiate recovery actions in order to take the system back to the region of required performance and prevent further degradations of performance;

Region of unacceptable performance: this region of operation is to be avoided through the use of a FTCS. This region delimits also the danger zone of operation, where there is an high risk of failures and severe faults, and which should be always avoided.

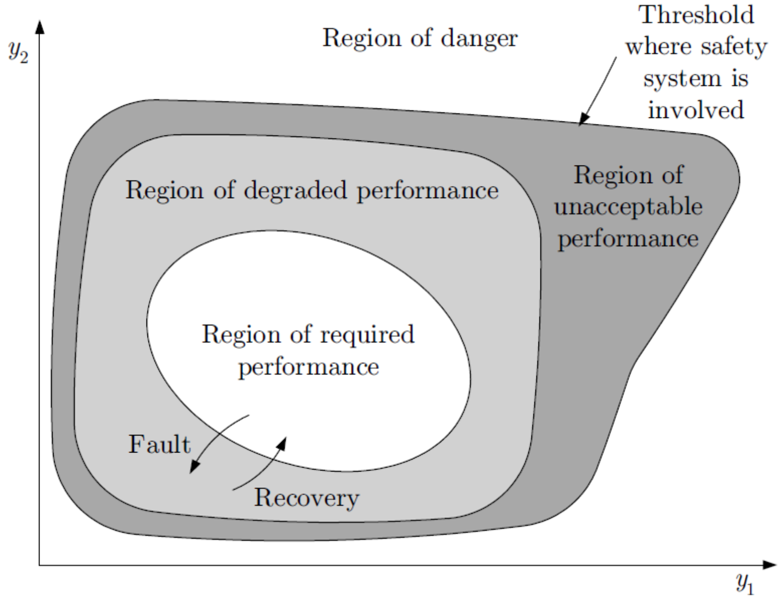

y_2
Region of danger
Threshold where safety system is involved
Region of degraded performance
Region of unacceptable performance
Region of required performance
Fault
Recovery
y_1

Figure 7.1 - Classification of the main regions of operation, based on FTC requirements (Blanke *et al.* 2006).

Generally speaking, FTCS can be classified into two broad categories: *Passive Fault Tolerant Control Systems* (PFTCS) and *Active Fault Tolerant Control Systems* (AFTCS) (Blanke *et al.* 2006; Zhang and Jiang 2008).

Passive Fault Tolerant Control (PFTC): in PFTC systems, the control law is fixed and does not change when a fault occurs. The control system is designed to be robust a priori against a set of limited (presumed) faults. A PFTC system is implemented with a constant feedback controller which is designed carefully using robust control techniques, with which the controller is designed such that the closed-loop system remains insensitive to system uncertainties and disturbances. When the effects of faults are similar to those of uncertainties or disturbance, it can be assured that the robust controller is insensitive to the faults. This approach requires neither FDD schemes nor controller reconfiguration, and is therefore computationally more attractive. Its applicability, however, is very restricted due to its serious disadvantages.

One of disadvantages of PFTC is that sometimes the fault is not incipient and has significant effect on the performance of the system, or it is not possible to design a controller to be robust to set of faults. Hence, a fault estimation scheme is needed to detect and identify the fault. Hence, in order to achieve such robustness to faults, usually a very restricted subset of the possible faults can be considered, when the effects of faults are similar to those of modelling errors and disturbances. Thus, often only faults that have a small effect on the behaviour of the system can be treated in this way, *i.e.* the passive approach has only limited fault-tolerant capabilities.

Since the control law is not changed when faults occur, the system is able to achieve its control goal, in general, only for objectives associated with a low level of performance (sometimes this is called the conservative approach). Moreover, achieving increased robustness to certain faults is only possible at the expense of decreased nominal performance. Since faults are effects that happen very rarely it is not reasonable to significantly degrade the fault-free performance of the system only to achieve some insensitivity to a restricted class of faults.

Active Fault Tolerant Control (AFTC): in general, an AFTC system reacts actively to faults and it is able to accommodate faults by reconfiguring the control actions so that the stability and acceptable performance of the entire system can be preserved. In certain circumstances, degraded performance may have to be accepted. Additionally, AFTC systems prevent faults in subsystems from developing into failures of the system. AFTC systems react to faults actively to recover or at least approximate the performance of the faulty system to the performance of the healthy system using the FDD results (Blanke *et al.* 2006; Zhang and Jiang 2008).

In such control systems, the controller compensates for the impacts of the faults either by selecting a predesigned control law or by designing a new one on-line. To achieve a successful control system reconfiguration, both approaches rely heavily on real-time FDD schemes to provide the most up-to-date information about the true status of the system, by detecting and localizing the faults that eventually occur in the system. The estimated faults (if available) are subsequently passed to a Reconfiguration Mechanism (RM) that changes the parameters and/or the structure of the controller in order to achieve an acceptable post-fault system performance.

When severe faults such as the complete failure of actuators or sensors breaks the control loops, it is necessary to use a different set of inputs or outputs for the control task. In these cases, in contrast to PFTC, AFTC consists of finding and implementing a new control structure in response to the occurrence of a severe fault. After selecting the new control configuration, new controller parameters would be found. The restructuring/redesign of controller is carried out automatically during the operation of the system, without needing any external intervention.

Therefore, the main goal in a fault tolerant control system is to design a controller with a suitable structure to achieve stability and satisfactory performance, not only when all control components are functioning normally, but also in cases when there are malfunctions in sensors, actuators, or other system components.

Due to their improved performance and their ability to deal with a wider class of faults, the active methods for FTC have gained much more attention in the literature than the passive ones.

As shown in Fig. 7.3, on-line fault accommodation and on-line controller-reconfiguration are usually used in AFTC framework. The controllers of AFTC are generally variable in parameter or even structure. The key issues in AFTC include the design of (Zhang and Jiang 2008):

1. A reconfigurable controller, which can be easily reconfigured;
2. A FDD scheme which is sensitive to faults and robust to model uncertainty, operating condition variations and external disturbances, providing as precisely as possible the information about an occurred fault (time, type and magnitude;
3. A controller reconfiguration mechanism, which compensates the fault-induced changes in the system and leads as closely as possible to the recovery of the pre-fault closed-loop system performance and stability in the presence of uncertainties, disturbances and time-delays in FDD, within the given constraints of control inputs and system states;
4. A command/reference governor.

It should be noted that the inclusion of both a FDD module and a reconfigurable controller with its reconfiguration mechanism within the system structure is the main difference between the active and the passive FTC system. In Fig. 7.2, the architecture of an AFTC is depicted.

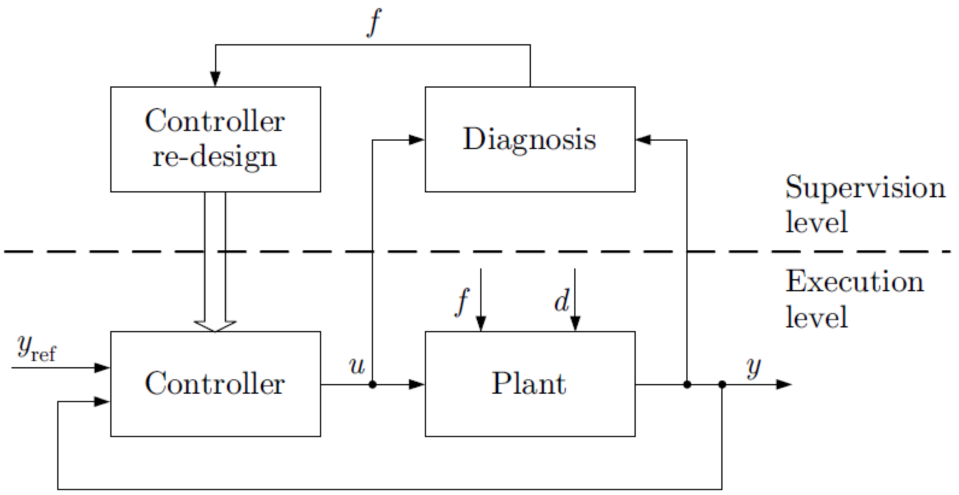


Figure 7.2 - Architecture of an active fault tolerant control (Blanke *et al.* 2006).

These elements of the AFTC design are highlighted also in Fig. 7.3. In the FDD module, any fault in the system should be detected and isolated as quickly as possible, and fault functions and post-fault system models need to be estimated on-line in real-time. Based on the on-line information on the post-fault system model, the control should be automatically reconfigured to maintain stability, desired dynamic performance and steady-state performance. In addition, in order to ensure the closed-loop system to track a command input trajectory in the event of faults, a reconfigurable feed-forward controller often needs to be synthesized. To avoid potential actuator saturation and to take into consideration the degraded performance after fault occurrence, in addition to a reconfigurable controller, a command/reference governor may also need to be designed to adjust command input or reference trajectory automatically.

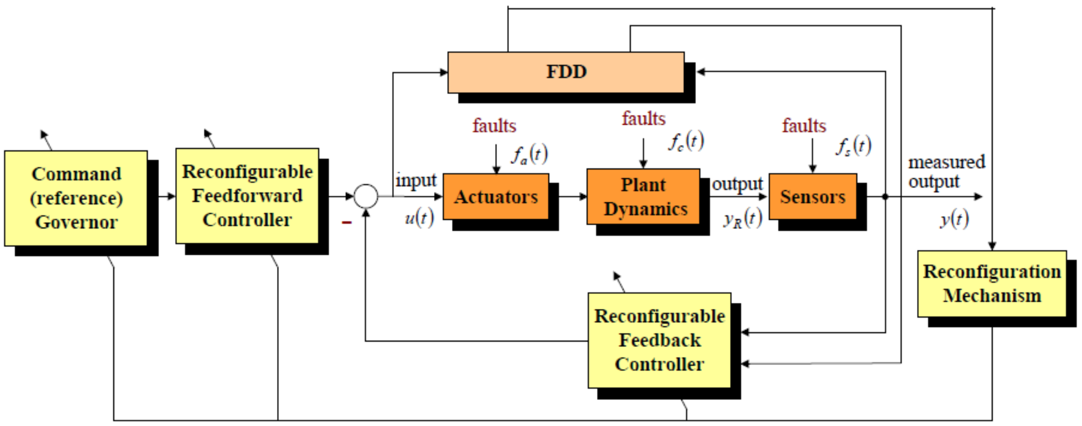

FDD
faults
$f_a(t)$
faults
$f_c(t)$
faults
$f_s(t)$
Command (reference) Governor
Reconfigurable Feedforward Controller
-
input
$u(t)$
Actuators
Plant Dynamics
output
$y_R(t)$
Sensors
measured output
$y(t)$
Reconfigurable Feedback Controller
Reconfiguration Mechanism

Figure 7.3 - Schematic structure of AFTC systems (Sun 2013).

There are different classifications for active fault tolerant control methods in the literature. According to (Lunze and Richter 2008), active methods can be classified into four major groups: physical redundancy, learning control, projection-based methods and on-line automatic controller redesign methods, as shown in Fig. 7.4. The latter method is concerned with defining new controller parameters or control law, known as a reconfigurable controller. In the projection based method, a set of controller are designed in advance and the system switches automatically between them such that an acceptable degree of performance of the system at the presence of faults is preserved. In safety-critical applications, the actuators and sensors are duplicated. When a fault happens, a simple decision algorithm switches the controller from a faulty component to a healthy one. This fault tolerant control method is known as physical redundancy. In the learning control method, the classical control techniques are combined with learning control method. Basically, a fast component, *e.g.* a Kalman filter, is used to estimate a changing condition quickly, then a slower learning component is employed to store previous knowledge to use it again in the future.

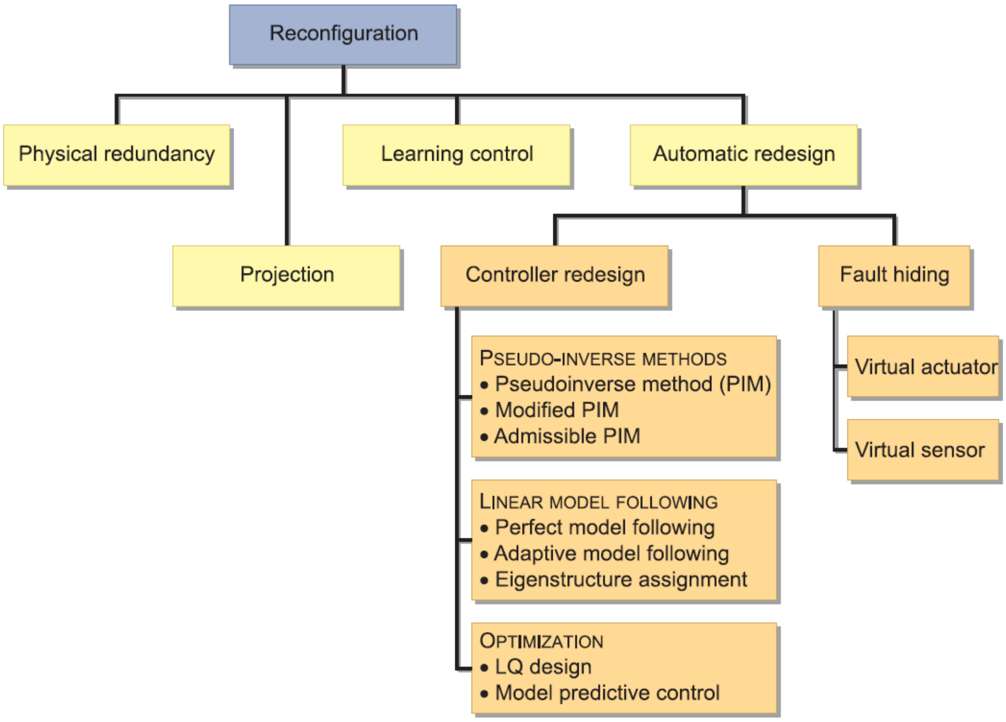


Figure 7.4 - Classification of control reconfiguration methods (Lunze and Richter 2008).

It is worth observing that, as already stated above, not only the parameters of the controllers need to be recalculated in the control redesign procedure, but also the structure of the new controllers might be changed (*i.e.* restructurable controller). This controller restructuring also uses alternative input and output signals in the new controller configuration. Afterwards, a new control law has to be designed on-line. The restructuring of the controller is necessary after occurrence of severe faults (*i.e.* complete component failures) that lead to serious changes of the plant behaviour. The necessity of control restructuring is apparent if actuator or sensor failures are contemplated. The total failure of these components leads to a breakdown of the control loop. Hence, a simple adaptation of the controller parameters to a new situation is no longer possible and hence alternative sensors or actuators have to be taken into account, preferably the ones that have similar interactions with the system and not being under the influence of a fault. Therefore, it is possible to design a new controller that satisfies the performance specification of the nominal system. This is possible only in case of presence of redundant actuators. From this point of view, the two principal ways of controller redesign can be differentiated as *fault accommodation* versus *system reconfiguration* (Blanke *et al.* 2006).

7.2 Control Reconfiguration and Fault Accommodation

Fault accommodation means to control the variables of interest in the faulty system, under the constraints associated with the faulty components. The control law is changed in response to fault, without switching off any system component. The sets of manipulated and measured signals remain unchanged and the control adjustment is limited to the controller dynamics. In fault accommodation, faulty components are still kept in operation thanks to an adapted control law. On the other hand, system reconfiguration means to switch off faulty components, and eventually switch on healthy redundant ones, and accordingly changing the control law so that controlling the resulting new system can achieve the control objectives under the set of constraints associated with the reconfigured system. Both the controller dynamics and the closed-loop structure, and possibly also the reference signal, are changed. In system reconfiguration, faulty components are no longer employed (Blanke *et al.* 2006; Zhang and Jiang 2008).

Hence, the main distinction between reconfigurable control and fault accommodation is that reconfigurable control includes modifications of the closed-loop signal structure, which are excluded in fault accommodation as shown in Figs. 7.5 and 7.6. In other words, the use of different input and output signals is allowed in reconfigurable control. Reconfiguration in fault-tolerant control exploits the presence of redundancy in the controlled system. Reconfigurable control is based on models of the fault-free and the faulty system, where the latter is provided by the fault diagnosis component or by self-diagnosing actuators and sensors. The model of the faulty plant must express all redundancies so that automatic reconfiguration methods are enabled to exploit them.

Reconfigurable control must at least ensure that the reconfigured closed-loop system is stable in a suitable sense. Furthermore, it is desirable to recover the nominal closed-loop tracking and performance properties as far as possible. The exact recovery of these properties is typically possible only in the presence of physical redundancy in the system. In numerous technological systems, physical redundancy is not available due to its cost. The appearance of actuator faults typically turns the nominal system into an under-actuated system, whereas the nominal system may be either fully actuated or under-actuated.

Generally speaking, the need for reconfigurable control arises whenever the controlled plant abruptly undergoes substantial structural changes. In the projection-based method, a set of controller are designed in advance (off-line) and the system switches automatically between them such that a sacrificed degree of performance of the system at the presence of faults are preserved.

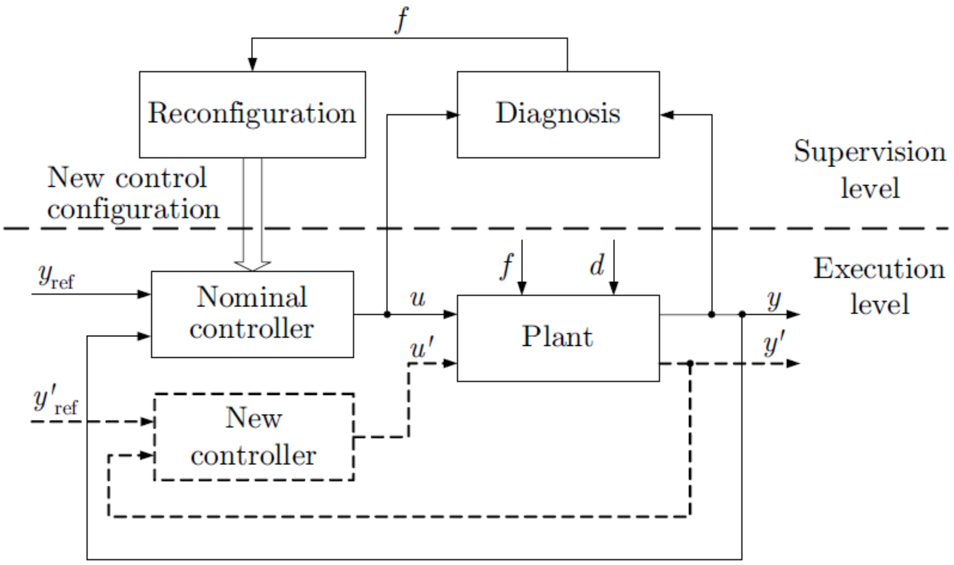


Figure 7.5 - Control reconfiguration (Blanke *et al.* 2006).

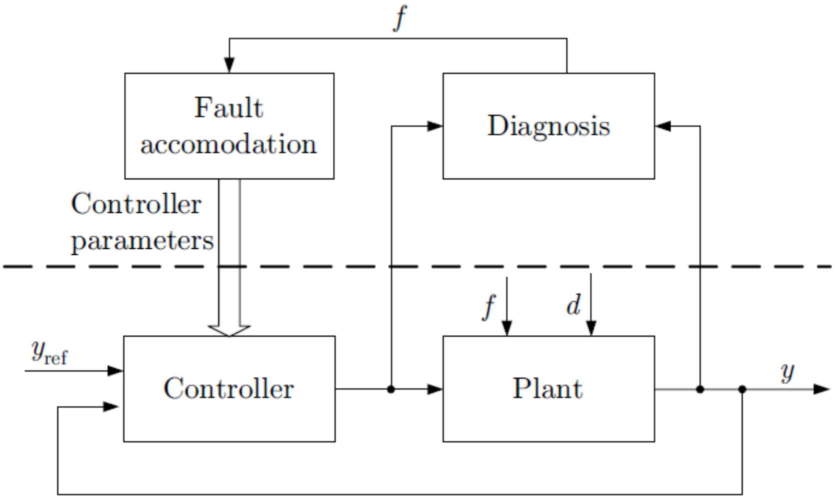


Figure 7.6 - Fault accommodation (Blanke *et al.* 2006).

7.3 Reconfiguration Mechanism

Under the FTC framework, once the FDI module indicates which sensor or actuator is faulty, the fault magnitude has been estimated by the FE module, a new control law will be set up in order to compensate for the fault effect on the system. As sensor and actuator faults have different effects on the system, the control law should be modified according to the nature of the fault, as shown is Fig. 7.7. In this dissertation, only a single (sensor or actuator) fault is assumed to occur at a given time, because simultaneous faults can hardly be isolated. In many applications, the presence of multiple faults, either simultaneous or subsequent, is rare.

Considering the occurrence of sensor faults, since they will not change the system dynamics, it is not necessary to redesign the controller to exclude the faulty sensor, if the faulty output can be corrected when a fault occurs. Therefore, the most commonly used approach is *sensor fault masking* using a software sensor (Wu *et al.* 2006; Zhang 2009), or equivalently using virtual sensors (Blanke *et al.* 2006; Ponsart *et al.* 2010; Nazari *et al.* 2013). The idea of the software or virtual sensor is to estimate the sensor fault or the real system outputs, and then feed the derived fault-free output signals to the controller, thereby decouple the effects of the faults in the feedback loop (Bennett 1998; Blanke *et al.* 2006; Wu *et al.* 2006; Gao *et al.* 2007; Gao and Ding 2007; Rothenhagen and Fuchs 2009; Zhang 2009; Ponsart *et al.* 2010). The key issue of this strategy is the requirement of fast and accurate fault estimation, which should be robust to uncertainty and disturbances. The compensation law would be in the following additive form:

$$y_{k,acc}(t) = y_k(t) - \hat{F}_{x_k}(t)$$

where $y_{k,acc}(t)$ is the *k-th* accommodated sensor measurement, $y_k(t)$ is the corresponding sensor measurement in faulty condition and $\hat{F}_{x_k}(t)$ is the fault estimate used to compensate for the impact of the actual sensor fault $F_{x_k}(t)$, as defined in Subection 3.5.2. In the normal or healthy condition, $\hat{F}_{x_k}(t)$ is identical or close to zero. Once a sensor fault is detected, $\hat{F}_{x_k}(t)$ should be suitable to compensate for the effect of the corresponding actual sensor fault. It is worth noting that this direct accommodation of an occurred sensor fault is valid also in case of failure of the sensor itself, and in this case the measurement estimation provided by the adaptive fault estimation filter will be equivalent to the output of an analytical virtual sensor measuring the same state variable through the adaptive estimation. This aspect remove the necessity of having two or more distinct sensors performing the same operation or measurement in order to have a certain level of redundancy in case of fault occurrence.

On the other hand, considering the occurrence of soft (*i.e.* not severe) actuator faults, the most used compensation law would be in the following additive form (Noura *et al.* 2000; Zhang and Jiang 2008):

$$u_{acc}(t) = u(t) - \hat{f}_u(t)$$

where $u_{acc}(t)$ is the accommodated control input, $u(t)$ is the actuated input in faulty condition and $\hat{f}_u(t)$ is the input term used to compensate for the impact of the actual fault $f_u(t)$, as defined in Subsection 3.5.1. In the normal or healthy condition, $\hat{f}_u(t)$ is identical or close to zero. Once an actuator fault is detected, $\hat{f}_u(t)$ should be suitable to compensate for the effect of the corresponding actual fault.

However, in case of total loss of effectiveness of an actuator (*i.e.* an actuator failure), a control system reconfiguration is required in order to recover the loss of actuation using one or more redundant actuators (Patton 1997a; Blanke *et al.* 2006). Changes in the controller parameters may not be enough to accommodate the total loss of actuator function and the controller structure or controller strategy must be redesigned on-line. The restructure or redesign is often done off-line, so that pre-computed redundant control laws and hardware systems can be selected once an actuator is known to have failed. According to whether or not the reconfigurable controller is calculated on-line, the AFTC can be classified according to whether a pre-calculated controller approach is used or whether the controller is designed on-line (Patton 1997a; Zhang and Jiang 2008).

8 FTC IMPLEMENTATION IN THE SATELLITE ADCS

This chapter presents the implementation of an AFTC scheme for the active accommodation of an occurred sensor or (soft) actuator fault or for the controller reconfiguration in case of failure of an attitude control actuator, realized by exploiting the available fourth redundant actuator present in the ACS of the spacecraft. The implementation of a standard Sliding Mode Controller (SMC) for attitude stabilization and control is presented as well.

8.1 Sliding Mode Control

Attitude control exploits the measurements and a reference to calculate a torque demand that will make the measured state (either attitude or angular velocity) equal to the reference. Considering an attitude reference, the satellite body coordinate system and a target coordinate system will be aligned when zero rotation is needed to align the satellite body system with the axes of the target system. On the other hand, when an angular velocity reference has to be followed, the control torques should keep the error on the spacecraft angular velocity equal to zero.

As already stated, the nominal Attitude Control System (ACS) is realized by considering the array of four redundant reaction wheels arranged in a tetrahedral configuration and implementing a simple Sliding Mode Control (SMC) in order to track a desired attitude reference (Young and Utkin 1999; Jiang *et al.* 2004; Bastoszewicz and Zuk 2010; Ibrahim *et al.* 2012).

Sliding mode control has been recognized as one of the efficient tools to design robust controllers for complex high-order nonlinear dynamic system operating under various conditions. The main advantages of SMC, as a Variable-Structure-Control (VSC) approach, are its fast dynamic response, robustness, simplicity in design and implementation. However, SMC presents a main drawback, the well known chattering phenomenon.

Essentially, sliding mode control utilizes discontinuous feedback control laws to force the system state to reach, and subsequently to remain on, a specified surface within the state space (the so-called sliding or switching surface). The system dynamic when confined to the sliding surface is described as an ideal sliding motion and represent the controlled system behaviour.

The advantages of obtaining such a motion are twofold: firstly the system behaves as a system of reduced order with respect to the original plant; and secondly the movement on the sliding surface of the system is insensitive to a particular kind of perturbation and model uncertainties. This latter property of invariance towards so-called matched uncertainties is the most distinguish feature of sliding mode control and makes this methodology particular suitable to deal with uncertain nonlinear systems. However, matched uncertainties include only uncertainties (and hence faults) on the channels associated with the control inputs. Hence, SMC schemes inherently have Passive Fault Tolerant Control (PFTC) capabilities, but only in case of actuator faults. Therefore, in case of sensor faults, an alternative fault tolerant scheme needs to be exploited to guarantee the desired performances. The attitude dynamic equations of a rigid spacecraft are given by Eq. (2.46), which for simplicity can be written for control design purpose as

$$\dot{\omega} = f(x) + bu + d \tag{8.1}$$

where

$$f(x) = -\mathbf{I}_{sat}^{-1} S(\boldsymbol{\omega}_{ib}^b)(\mathbf{I}_{sat}\boldsymbol{\omega}_{ib}^b + \mathbf{h}_w^b)$$

$$b = \mathbf{I}_{sat}^{-1}\mathbf{A}_w$$
$$\omega = \begin{bmatrix}\omega_1 & \omega_2 & \omega_3\end{bmatrix}^T = \boldsymbol{\omega}_{ib}^b \in \mathbb{R}^{3\times1}$$
$$u = \begin{bmatrix}u_1 & u_2 & u_3 & u_4\end{bmatrix}^T = \mathbf{T}_{ctrl} \in \mathbb{R}^{4\times1}$$
$$d = \begin{bmatrix}d_1 & d_2 & d_3\end{bmatrix}^T = \mathbf{T}_{ext}^b = \mathbf{T}_{gg}^b + \mathbf{T}_{aero}^b \in \mathbb{R}^{3\times1}$$

Moreover, since the gravitational disturbance torque is assumed to be exactly modelled in this thesis, the term $\mathbf{T}_{gg}^b$ could be considered directly in the term $f(x)$, and the disturbance d assumed consisting only of the aerodynamic disturbance torque.

The quaternion error $\bar{\mathbf{q}}_e = \begin{bmatrix}\mathbf{q}_e & q_{e4}\end{bmatrix}^T$, which denotes the relative attitude error from the desired reference frame to the body-fixed frame, is defined as

$$\bar{\mathbf{q}}_e = \bar{\mathbf{q}}_r^* \otimes \bar{\mathbf{q}}_{star} = \begin{bmatrix} q_{e1} \\ q_{e2} \\ q_{e3} \\ q_{e4} \end{bmatrix} = \begin{bmatrix} q_{star4} & q_{star3} & -q_{star2} & q_{star1} \\ -q_{star3} & q_{star4} & q_{star1} & q_{star2} \\ q_{star2} & -q_{star1} & q_{star4} & q_{star3} \\ -q_{star1} & -q_{star2} & -q_{star3} & q_{star4} \end{bmatrix} \begin{bmatrix} -q_{r1} \\ -q_{r2} \\ -q_{r3} \\ q_{r4} \end{bmatrix} \tag{8.2}$$

where $\bar{\mathbf{q}}_{star} = \begin{bmatrix}\mathbf{q}_{star} & q_{star4}\end{bmatrix}^T$ describes the attitude measured by star sensors, $\bar{\mathbf{q}}_r = \begin{bmatrix}\mathbf{q}_r & q_{r4}\end{bmatrix}^T$ describes the desired target attitude. Since a LEO satellite is considered in this thesis, the satellite is requested to maintain a fixed attitude (either a nadir-pointing or a generic Earth-pointing attitude) with respect to the orbital frame. Hence, the two quaternions $\bar{\mathbf{q}}_{star}$ and $\bar{\mathbf{q}}_r$ describe the satellite attitude with respect to the orbital frame. On the other hand, in case a fixed inertial pointing (*e.g.* the pointing of a specific star) is needed, the two quaternions $\bar{\mathbf{q}}_{star}$ and $\bar{\mathbf{q}}_r$, and the kinematic equations should be expressed with respect to the inertial frame.

The error $\boldsymbol{\omega}_e$ represents the difference between the actual angular velocity $\boldsymbol{\omega}_s$ of the spacecraft (*i.e.* the velocity measured by sensors expressed with respect to the orbital frame) and a desired reference (target) angular velocity $\boldsymbol{\omega}_r$, as follows:

$$\boldsymbol{\omega}_e = \boldsymbol{\omega}_s - \boldsymbol{\omega}_r \tag{8.3}$$

Considering a desired angular velocity $\boldsymbol{\omega}_r$ equal to zero for a three-axis stabilization at a constant desired attitude with respect to the orbital frame, it results that $\boldsymbol{\omega}_e = \boldsymbol{\omega}_s = \boldsymbol{\omega}_{ob}^b$ and the error rate dynamic equation is the same as in Eqs. (2.46) and (8.1). For any given two groups of quaternion, the relative attitude error can be obtained by

$$\dot{\bar{\mathbf{q}}}_e = \begin{bmatrix} \dot{\mathbf{q}}_e \\ \dot{q}_{e4} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} q_{e4}\mathbf{I}_3 + S(\mathbf{q}_e) \\ -\mathbf{q}_e^T \end{bmatrix} \boldsymbol{\omega}_e \tag{8.4}$$

Then, the error state vector $(\bar{\mathbf{q}}_e, \boldsymbol{\omega}_e)$ is fed to the controller to compute the control signal command. The designed sliding mode controller aims to drive the attitude states $(\bar{\mathbf{q}}_s, \boldsymbol{\omega}_s)$ from the initial state $(\bar{\mathbf{q}}_s(0), \boldsymbol{\omega}_s(0))$ to the desired states $(\bar{\mathbf{q}}_d, \boldsymbol{\omega}_d)$ with the following assumptions:

1. The measures of the unit quaternion $\bar{\mathbf{q}}_s$ and the body angular velocity $\boldsymbol{\omega}_s$ are available in the feedback control design;
2. The initial angular velocity of the satellite is zero;
3. The external disturbance vector d is assumed to be bounded and to satisfy the condition $\|d(t)\| < d_{\max}$. This assumption holds for the considered external disturbance torques (both the gravitational and aerodynamic ones) acting on a satellite.

8.1.1 Sliding Surface Design

The design of a sliding mode controller involves designing of a sliding surface that represents the desired stable dynamics and a control law that makes the designed sliding surface attractive (Young, Utkin 1999; Jiang *et al.* 2004; Bastoszewicz and Zuk 2010; Ibrahim *et al.* 2012). The phase trajectory of a sliding mode controller can be investigated in two parts, representing two modes of the system. The trajectory starting from a given initial condition off the sliding surface tend towards the sliding surface. This phase is known as reaching or hitting phase, and during this phase the system is sensitive to disturbances. When the hitting of the sliding surface occurs, the sliding phase starts, and during this phase the trajectories are insensitive to parameter variations and disturbances. A linear sliding surface has been defined for the considered spacecraft attitude control, defined in vector form as

$$\mathbf{s} = \boldsymbol{\omega}_e + c\mathbf{q}_e \tag{8.5}$$

where c is a strictly positive real constant determining the slope of the sliding surface $\mathbf{s}(t) = \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}^T \in \mathbb{R}^{3\times 1}$, which divides the state space into two parts.

As previously stated, the quaternion parameterization of the satellite attitude $\bar{\mathbf{q}}_e(t)$ and $-\bar{\mathbf{q}}_e(t)$ represent identical physical tracking error rotations. The former one gives the shortest angular distance to the sliding manifold, whilst the latter one gives the longest distance. A modification to the sliding surface can be implemented, like in Crassidis *et al.* (1999), to guarantee that the satellite will follow the shortest angular path to reach the sliding manifold, thus reducing the required amount of control torque. This modified sliding surface is defined as

$$\mathbf{s} = \boldsymbol{\omega}_e + c \cdot sgn(q_{e4})\mathbf{q}_e \tag{8.6}$$

with the sign term $sgn(q_{e4})$ used to drive the system to the desired trajectory in the shortest way.

8.1.2 Control Law Design

The sliding mode control law can be divided into two main parts (Young and Utkin 1999; Ibrahim *et al.* 2012):

$$\mathbf{u}(t) = \mathbf{u}_{eq}(t) + \mathbf{u}_d(t) \tag{8.7}$$

The component $\mathbf{u}_{eq}(t) = \begin{bmatrix} u_{1eq} & u_{2eq} & u_{3eq} & u_{4eq} \end{bmatrix}^T \in \mathbb{R}^{4\times 1}$ makes the sliding surface $s(t)$ invariant ($\dot{\mathbf{s}} = 0$ for $\mathbf{s} = 0$) and it is calculated by setting the derivative $\dot{\mathbf{s}}$ to zero considering $\mathbf{s}(t)$ to be zero. The component $\mathbf{u}_d(t) = \begin{bmatrix} u_{1d} & u_{2d} & u_{3d} & u_{4d} \end{bmatrix}^T \in \mathbb{R}^{4\times 1}$ is an extra control effort which forces the quaternion and angular velocity components to reach the sliding surface in a finite time despite of the presence of disturbances (attractive sliding surface with $\mathbf{s}^T\dot{\mathbf{s}} < 0$). This switching control law

$\mathbf{u}_d(t)$ is computed according to a discontinuous constant reaching law (Slotine and Li 1991; Bastoszewicz and Zuk 2010; Ibrahim *et al.* 2012) as

$$\mathbf{u}_d = b^{-1}(-\mathbf{K}_{SMC} \cdot sgn(\mathbf{s})) \tag{8.8}$$

where $\mathbf{K}_{SMC}$ is a proper positive definite gain vector and

$$\mathrm{sgn}(s) = \begin{cases} 1 & \text{for } s > 0 \\ -1 & \text{for } s < 0 \end{cases}$$

In order to compute the equivalent control law $\mathbf{u}_{eq}(t)$, it is set

$$\dot{\mathbf{s}} = \dot{\boldsymbol{\omega}}_e + c \cdot sgn(q_{e4})\dot{\mathbf{q}}_e = 0 \tag{8.9}$$

and, by substituting Eq. (8.1) in Eq. (8.9), it results

$$\mathbf{u}_{eq} = b^{-1}(-f - c \cdot sgn(q_{e4})\dot{\mathbf{q}}_e) \tag{8.10}$$

and, therefore

$$\mathbf{u} = b^{-1}(-f - c \cdot sgn(q_{e4})\dot{\mathbf{q}}_e - \mathbf{K}_{SMC} \cdot sgn(s)) \tag{8.11}$$

The control law of Eq. (8.11) has two design parameters $(c, \mathbf{K}_{SMC})$ that should be selected to provide stability and better performances. The slope c of the sliding surface is selected such that the system is stable during the sliding mode. A large enough discontinuous signal $\mathbf{K}_{SMC} \cdot sgn(s)$ is necessary to complete the reachability condition despite perturbations and disturbances, but as small as possible in order to limit the chattering phenomenon. According to the selected value of $\mathbf{K}_{SMC}$, the state trajectories will reach the sliding surface at different points with different reaching times, as shown in Fig. 8.1. For small values, the state trajectories take more time and long path to reach.

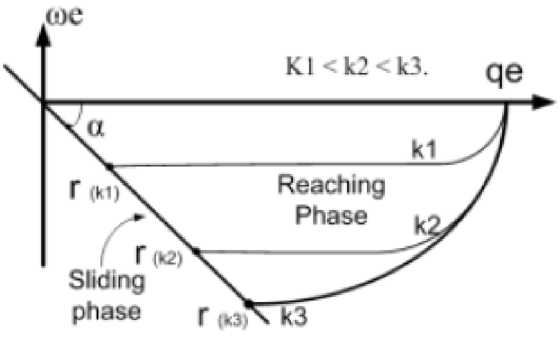


Figure 8.1 - Error phase-plane for different gain values (Ibrahim *et al.* 2012).

8.1.3 Chattering Avoidance and Control Input Saturation

The chattering phenomenon is generally perceived as motion which oscillates about the sliding manifold. Finite-frequency finite-amplitude oscillations are caused by an high-frequency switching of the sliding mode controller. A chattering control signal may cause possible damages to the actuators. A boundary layer method can be used to reduce the chattering. The basic idea is to replace the sign function by a saturation function (Dal and Teodorescu 2011; Ibrahim *et al.* 2012), defined as

$$sat(s_i,\varepsilon_i)=\begin{cases}1 & \text{for } s_i>\varepsilon_i\\ s_i/\varepsilon_i & \text{for } |s_i|\le\varepsilon_i \qquad i=1,2,3\\ -1 & \text{for } s_i<-\varepsilon_i\end{cases} \tag{8.12}$$

The larger the boundary layer width ε_i the smoother the control signal $\mathbf{u}$ (the discontinuous control term $\mathbf{u}_d$ is softened). However, it no longer drives the system to the origin, but only to within the chosen boundary layer instead. A very small thickness layer introduces again the chattering phenomenon.

Finally, since the considered reaction wheels are characterized by maximal values of the control torques they can effectively actuate, a saturation of the commanded control torque is considered. Hence, it results that the actual commanded control inputs are defined as

$$u_{ctrl,i}=\begin{cases}u_{\max} & \text{for } u_i\ge u_{\max}\\ u_i & \text{for } -u_{\max}<u_i<u_{\max} \qquad i=1,2,3,4\\ -u_{\max} & \text{for } u_i\le -u_{\max}\end{cases} \tag{8.13}$$

8.1.4 Controller Stability in Nominal Condition

Considering the sliding surface of Eq. (8.6) and the control law of Eq. (8.11), the system reaches to the sliding surface $\mathbf{s}=0$ in finite time. In fact, considering for the reaching phase the candidate Lyapunov function

$$V_1=\frac{1}{2}\mathbf{s}^T\mathbf{s} \tag{8.14}$$

with time derivative

$$\dot{V}_1=\mathbf{s}^T\dot{\mathbf{s}}=\mathbf{s}^T(\mathbf{d}-\mathbf{K}\cdot\operatorname{sgn}(\mathbf{s})) \tag{8.15}$$

the time derivative must be negative to make $\mathbf{s}(t)$ reach zero (attractive sliding surface, $\mathbf{s}^T\dot{\mathbf{s}}<0$). This is obtained for $\mathbf{K}_{SMC}\ge|\mathbf{d}_{\max}|$, *i.e.* the gain must be selected such that it compensate for the upper bound of the disturbance.

Moreover, the trajectory in the error state-space that slides on the sliding surface can be shown to be asymptotically stable by using the Lyapunov's direct method. Considering the candidate Lyapunov function

$$V_2=\frac{1}{2}\mathbf{q}_e{}^T\mathbf{q}_e \tag{8.16}$$

and substituting (8.4) into (8.6), the following kinematic equation for ideal sliding motion on the sliding manifold ($\mathbf{s}=0$) is obtained:

$$\dot{\mathbf{q}}_e=-\frac{1}{2}c\cdot|q_{e4}|\mathbf{q}_e+\frac{1}{2}S(\mathbf{q}_e)(-\mathbf{K}_{SMC}\cdot sgn(q_{e4})\mathbf{q}_e) \tag{8.17}$$

Substituting (8.17) into the time derivative of (8.16) leads to the expression

$$\dot{V}_2 = \frac{1}{2}\dot{\mathbf{q}}_e^T\mathbf{q}_e + \frac{1}{2}\mathbf{q}_e{}^T\dot{\mathbf{q}}_e = \mathbf{q}_e{}^T\dot{\mathbf{q}}_e = -\frac{1}{2}c\cdot\left|q_{e4}\right|\mathbf{q}_e^T\mathbf{q}_e \tag{8.18}$$

which is clearly negative if $c > 0$. Therefore, the attitude tracking error $\mathbf{q}_e$ tends to zero as $t \to \infty$, and since the motion is on the sliding surface defined by $\mathbf{s}(t) = 0$, it follows that also $\boldsymbol{\omega}_e$ tends to zero as $t \to \infty$.

8.2 Simulation Results

In the case of actuator and sensor fault or failure, a reconfiguration is exploited in order to reconfigure the system maintaining the nominal sliding mode controller and achieve a tolerable performance. The designed AFTC scheme contains a fault diagnosis block and a system reconfiguration block. First, the occurred fault is detected and isolated by means of the proposed bank of residual generators, then the isolated fault is estimated by means of dedicated adaptive fault estimation filters, and finally the system is reconfigured a proper way, as shown in the following simulation results. The following figures show the results obtained in each fault scenario by implementing the AFTC scheme.

In order to present how the degraded performance of the attitude control system subject to a fault can be restored to a desired acceptable level, the simulation results obtained in the considered fault scenarios will be shown. In general, the comments that will be made on the basis of the analysis of the presented results are in general valid for all the possible fault scenarios.

8.2.1 Controller Reconfiguration after Actuator Failure

In case of occurrence of a severe actuator fault, *i.e.* an actuator failure, the direct fault accommodation could be no more possible due to the complete loss of effectiveness of the component. Hence, the exploitation of redundant actuators could be the only practical way to restore the nominal system performances or, at least, maintain an acceptable level of performance.

With the assumption of only one fault occurring at a time, the redundancy of the actuators in the considered ACS configuration can be practically exploited in order to restore the degraded performances and guarantee the complete controllability of the satellite after the fault occurrence.

Therefore, a system reconfiguration is performed, and only the remaining healthy actuators are exploited in order to control the satellite attitude. In case of failure of one of the four actuators, the actuator configuration matrix A_w, *i.e.* the equation term b used in the relations of the sliding mode controller of Eq. (8.11), is switched to another one in which the column corresponding to the failed actuator has all zero elements. In this way, only the other actuators are exploited for control allocation, *i.e.* for the distribution of the desired total control effort among a redundant set of actuators.

The following matrices can be used in case of a specific actuator failure. The actuator configuration matrix $A_{w,0}$ corresponds to the nominal condition where all the actuators are properly working. On the other hand, each of the other matrices $A_{w,i}$ $i = 1,\dots,4$ corresponds to the condition of failure of the *i-th* actuator.

$$\mathbf{A}_{w,0} = \begin{bmatrix} \sqrt{1/3} & \sqrt{1/3} & -\sqrt{1/3} & -\sqrt{1/3} \\ \sqrt{2/3} & -\sqrt{2/3} & 0 & 0 \\ 0 & 0 & -\sqrt{2/3} & \sqrt{2/3} \end{bmatrix}$$

$$\mathbf{A}_{w,1} = \begin{bmatrix} 0 & \sqrt{1/3} & -\sqrt{1/3} & -\sqrt{1/3} \\ 0 & -\sqrt{2/3} & 0 & 0 \\ 0 & 0 & -\sqrt{2/3} & \sqrt{2/3} \end{bmatrix} \qquad \mathbf{A}_{w,2} = \begin{bmatrix} \sqrt{1/3} & 0 & -\sqrt{1/3} & -\sqrt{1/3} \\ \sqrt{2/3} & 0 & 0 & 0 \\ 0 & 0 & -\sqrt{2/3} & \sqrt{2/3} \end{bmatrix}$$

$$\mathbf{A}_{w,3} = \begin{bmatrix} \sqrt{1/3} & \sqrt{1/3} & 0 & -\sqrt{1/3} \\ \sqrt{2/3} & -\sqrt{2/3} & 0 & 0 \\ 0 & 0 & 0 & \sqrt{2/3} \end{bmatrix} \qquad \mathbf{A}_{w,4} = \begin{bmatrix} \sqrt{1/3} & \sqrt{1/3} & -\sqrt{1/3} & 0 \\ \sqrt{2/3} & -\sqrt{2/3} & 0 & 0 \\ 0 & 0 & -\sqrt{2/3} & 0 \end{bmatrix}$$

In case of failure of the second actuator, the attitude control system switches to the matrix $A_{w,2}$, and the failed actuator is excluded. In Fig. 8.2, it is shown the evolution of the commanded control input, depicted in black, and the actual evolution of the actuated control input, depicted in blue, in the case in which the initial nominal matrix $A_{w,0}$ is used. In this case, the failed actuator cannot satisfy the request of torque, and then the performances of the overall control system could result to be degraded.

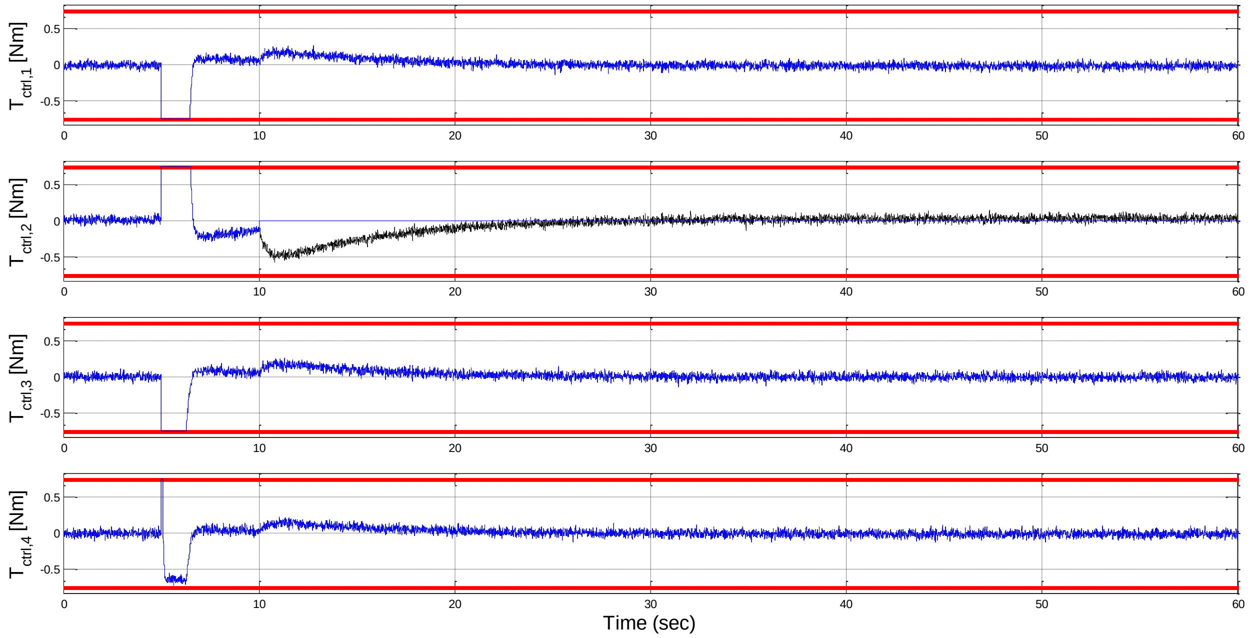


Figure 8.2 - Fault scenario n.5: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of failure of the actuator providing the control input $T_{ctrl,2}$.

After the fault detection, isolation and estimation, if it results that the estimated fault is equal to the commanded torque, it means that a failure has occurred. After the matrix switching, the failed line is excluded and only three actuators are exploited. Hence, the control allocation is varied. Fig. 8.3 shows how the actuated and commanded control torques are modified. It can be seen that the commanded torque for the failed actuator is set equal to zero, and the other actuated torques are properly changed after the reconfiguration. Moreover, Fig. 8.4 shows that the local residual of RB1 relative to the failed actuator presents a residual evolution that returns to zero after the exclusion of the failed control line, since also the corresponding commanded torque is set to zero and any discrepancy between the actual system and its model exploited by the residual filters is thus eliminated.

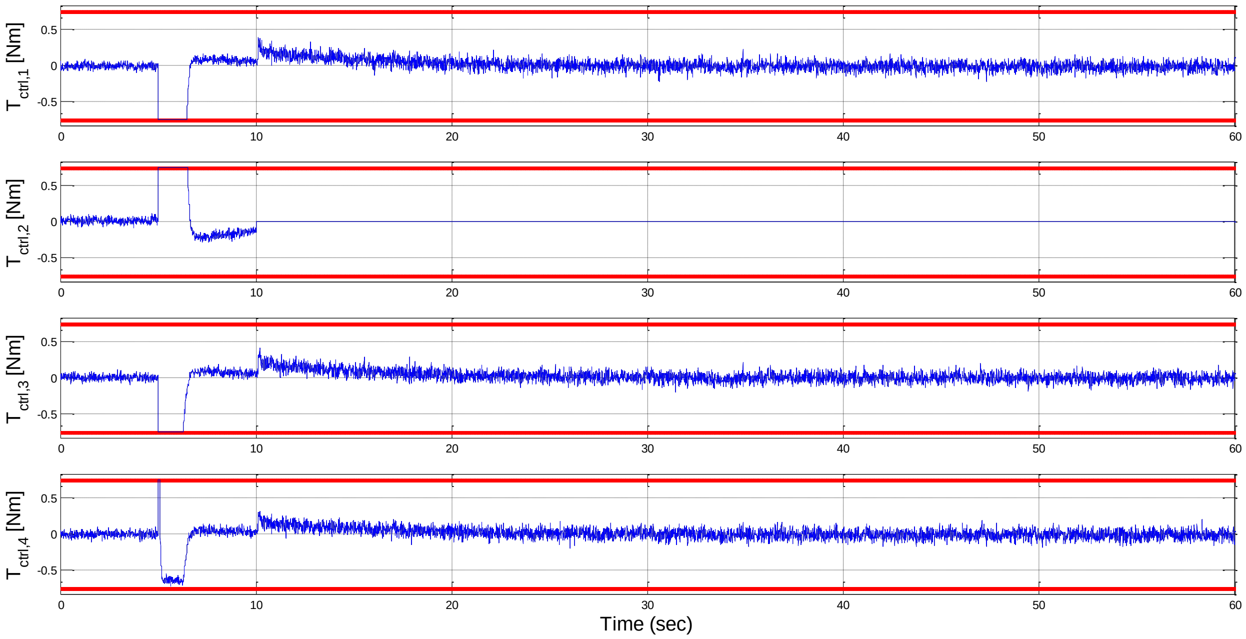


Figure 8.3 - Fault scenario n.5: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of failure of the actuator providing the control input $T_{ctrl,2}$. The second line is excluded from the control allocation.

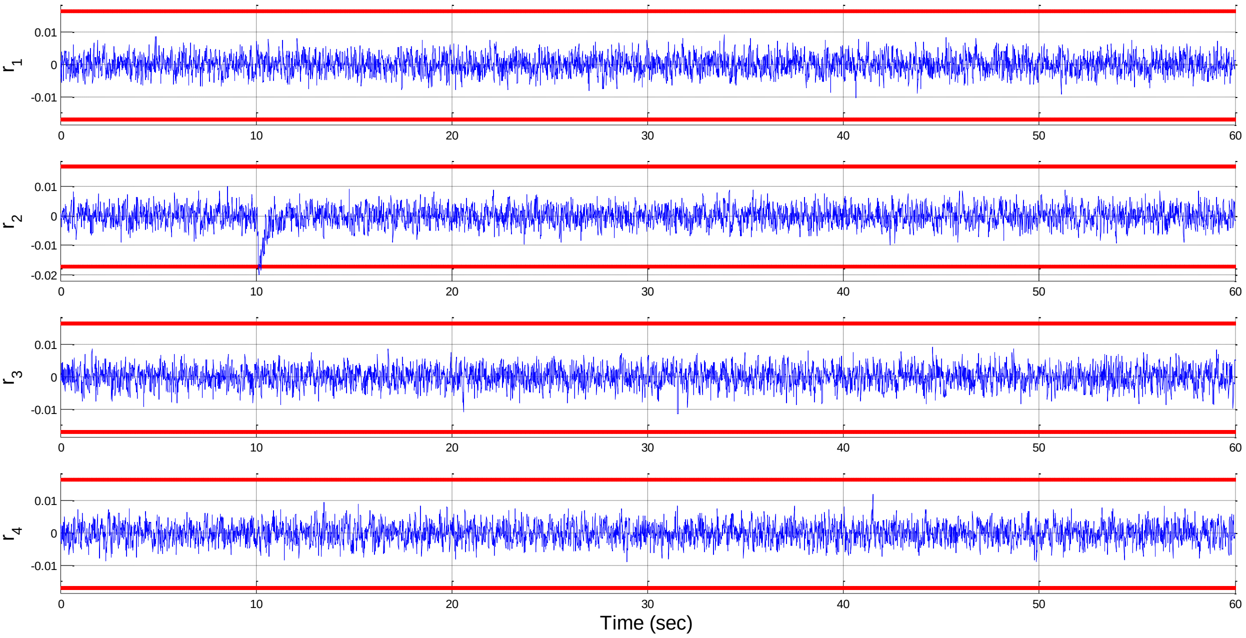


Figure 8.4 - Fault scenario n.5: local residuals of RB1 in case of failure of the actuator providing the control input $T_{ctrl,2}$ and subsequent control reallocation.

8.2.2 Fault Accommodation after Actuator and Sensor Faults in the Satellite ACS

On the other hand, if a soft fault occurs, affecting either an actuator or a sensor of the ADCS, a fault accommodation approach can be exploited. As already described in Section 7.3, the simplest way to realize the fault accommodation is to exploit the fault estimation as a compensation term to correct the faulty input or output signal.

Considering, for example, the first fault scenario, *i.e.* the step fault affecting the attitude control input $T_{ctrl,2}$, Figs. 8.5 and 8.6 show the evolution of the commanded and actuated control inputs without and with fault accommodation. It can be seen that the fault accommodation allows obtaining an actuated control input equal to the required nominal one as if the fault had not happened, due to the compensation of the fault effect on the control input itself by means of the estimated fault. The fault accommodation allows to avoid the changing of all the control inputs after the fault occurrence caused by the sliding mode controller reacting to the fault occurrence, and recover the nominal control input evolution and system behaviour.

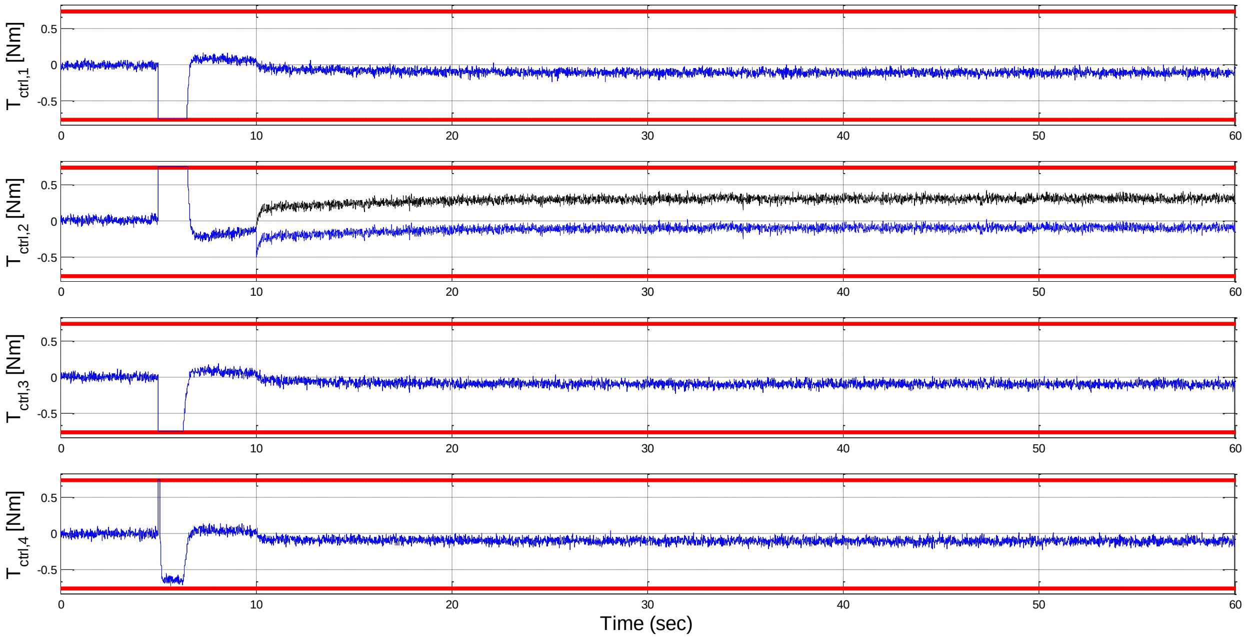


Figure 8.5 - Fault scenario n.1: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of step fault on the control input $T_{ctrl,2}$ and without fault accommodation.

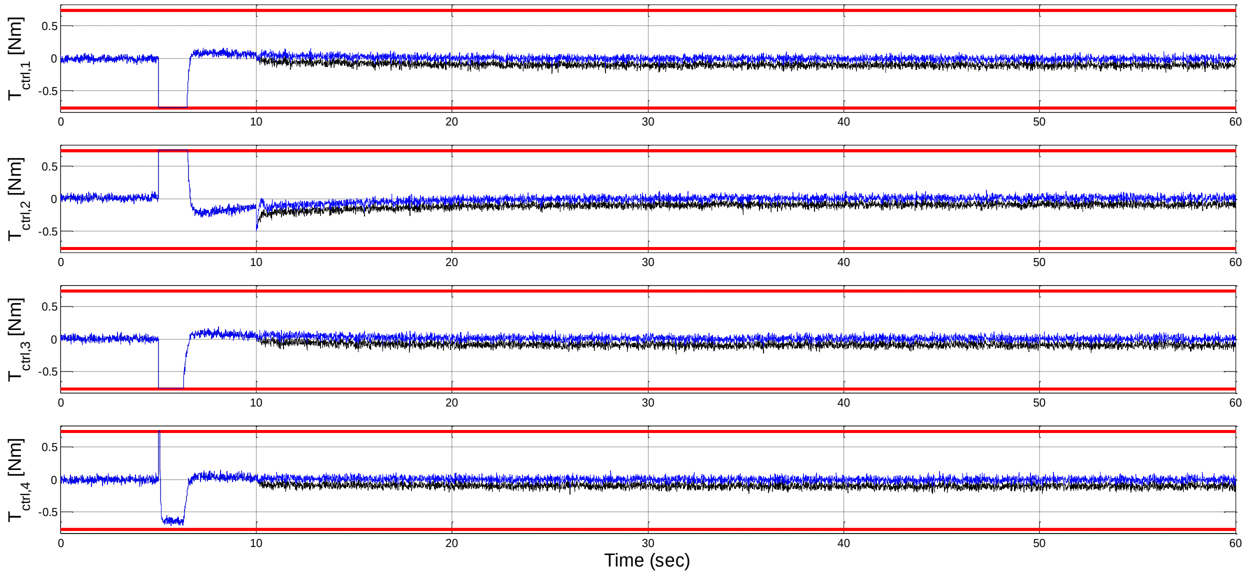


Figure 8.6 - Fault scenario n.1: faulty (black) and accommodated (blue) actuated control inputs T_{ctrl} in case of step fault on the control input $T_{ctrl,2}$ and with fault accommodation.

After the accommodation, the performances of the control system are restored and the satellite can reach and maintain the desired attitude with respect to the orbital reference frame, as shown for example in Figs. 8.7 and 8.8. In particular, the effects of the fault occurrence and fault accommodation are clearly from these two figures evident for the yaw angle of the satellite with respect to the orbital reference frame.

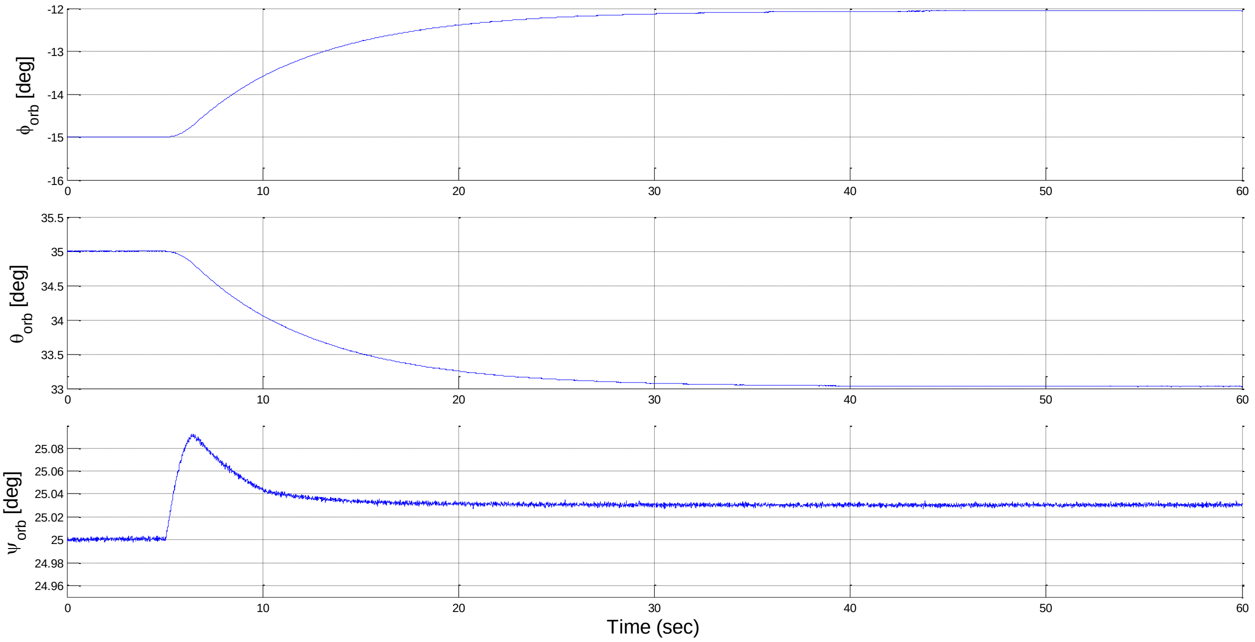


Figure 8.7 – Fault scenario n.1: Euler angles evolution in case of step fault on the control input $T_{ctrl,2}$ and without fault accommodation.

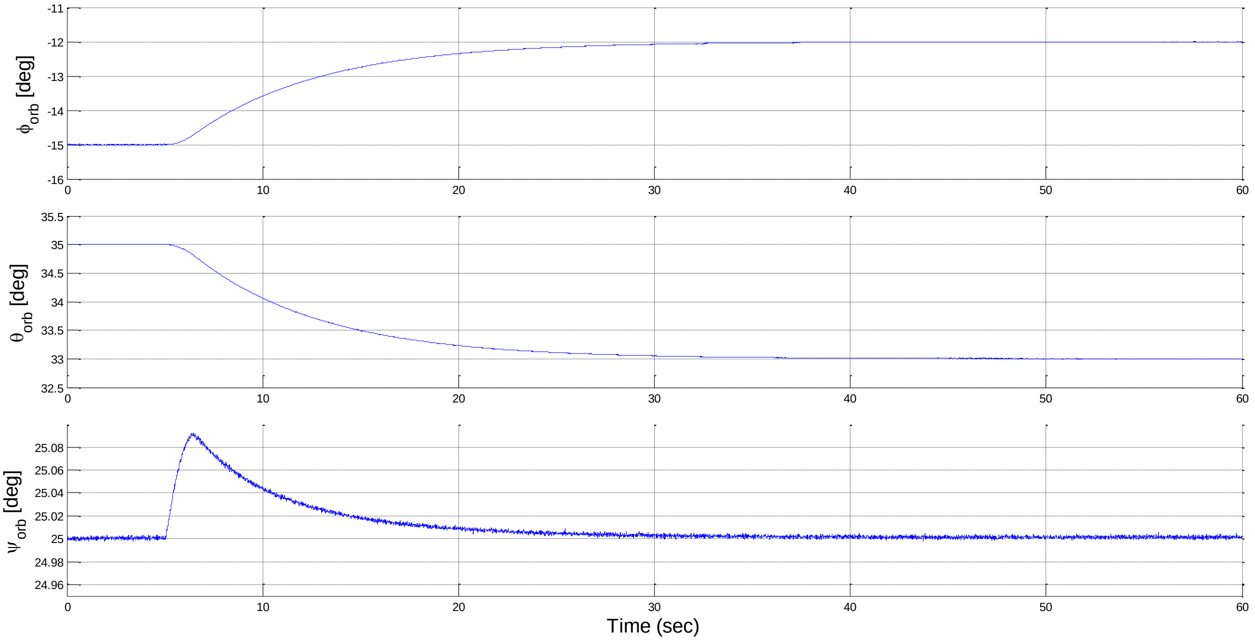


Figure 8.8 – Fault scenario n.1: Euler angles evolution in case of step fault on the control input $T_{ctrl,2}$ and with fault accommodation.

After the fault accommodation, even in this case the local residual of RB1 sensitive to the occurred fault returns to zero after an initial transient phase during which the fault is detected, isolated and the fault estimation is subsequently enabled.

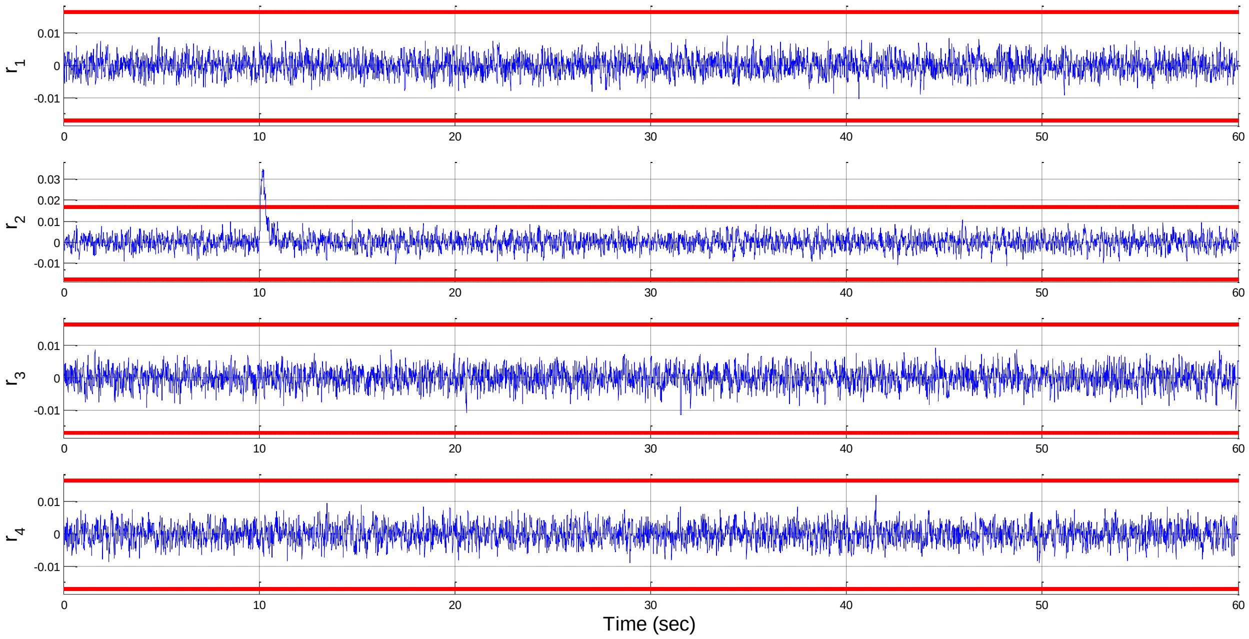


Figure 8.9 - Fault scenario n.1: local residuals of RB1 in case of step fault on the attitude control input $T_{ctrl,2}$ and subsequent fault accommodation.

For all the other considered scenarios concerning the occurrence of faults affecting a satellite actuator, very similar results can be obtained. Figs. 8.10 and 8.11 show the comparison between the commanded and actuated control inputs in case of occurrence of a sinusoidal fault and without and with fault accommodation, respectively. In case of fault accommodation, the effects of the occurred fault on the actuated control inputs are almost completely compensated, as shown in Fig. 8.11. Only a slight oscillation of the sensitive local residual of RB1 lasts in Fig. 8.12, because of the remaining fault effects due to the not null (but bounded) fault estimation error shown in Fig. 6.6 and the subsequent not perfect fault compensation. Anyway, the fault accommodation makes the residual sensitive to the occurred fault to return and remain within the selected thresholds.

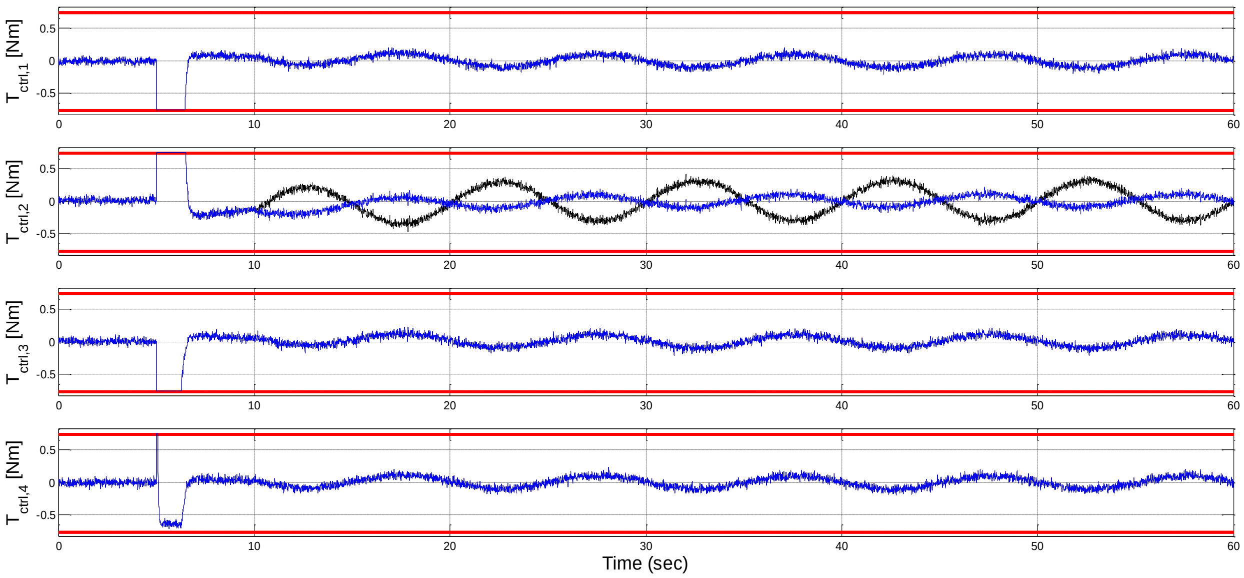


Figure 8.10 - Fault scenario n.2: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of sinusoidal fault on the control input $T_{ctrl,2}$ and without fault accommodation.

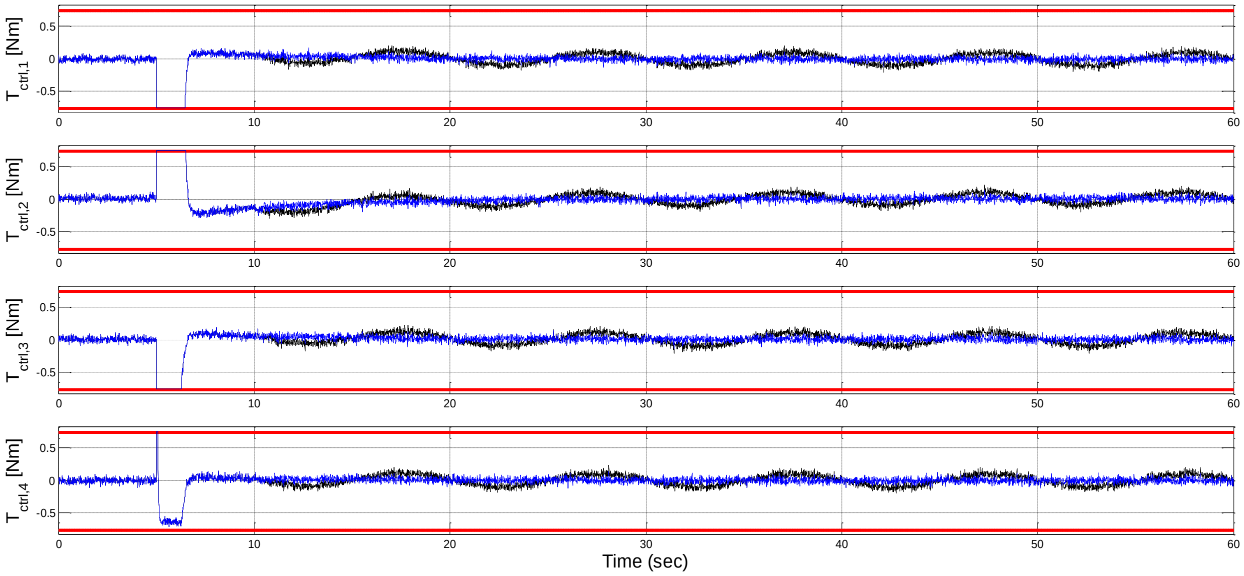


Figure 8.11 - Fault scenario n.2: faulty (black) and accommodated (blue) actuated control inputs T_{ctrl} in case of sinusoidal fault on the control input $T_{ctrl,2}$ and with fault accommodation.

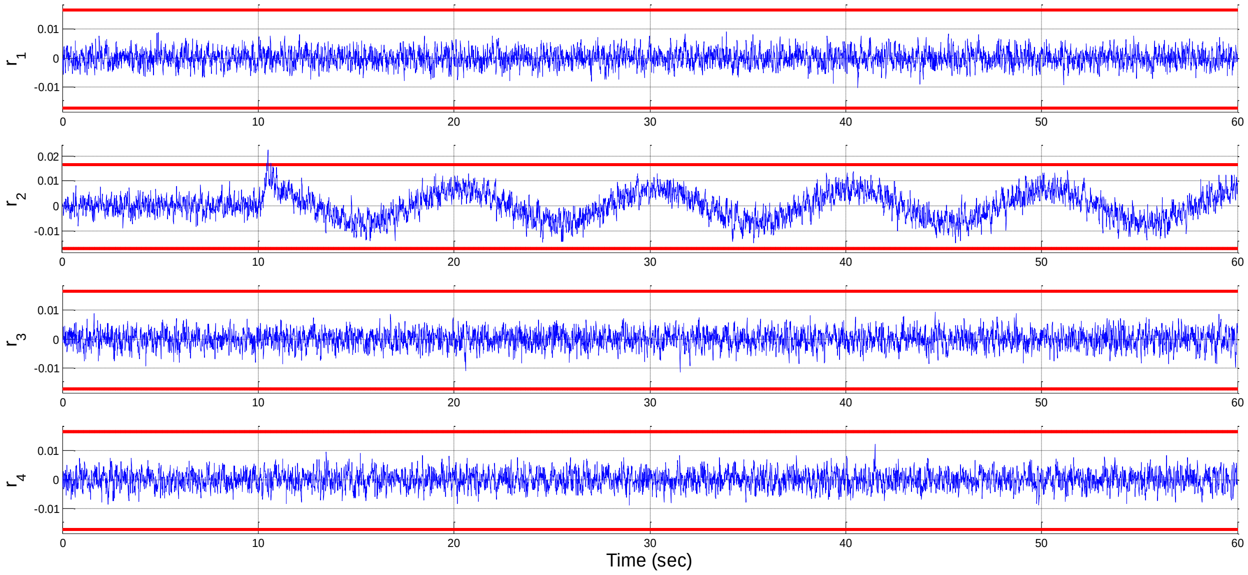


Figure 8.12 - Fault scenario n.2: local residuals of RB1 in case of sinusoidal fault on the attitude control input $T_{ctrl,2}$ and subsequent fault accommodation.

Figs. 8.13, 8.14 and 8.15 show results analogous to the case of the step actuator fault, but in case of a rectangular pulse fault affecting the second actuator.

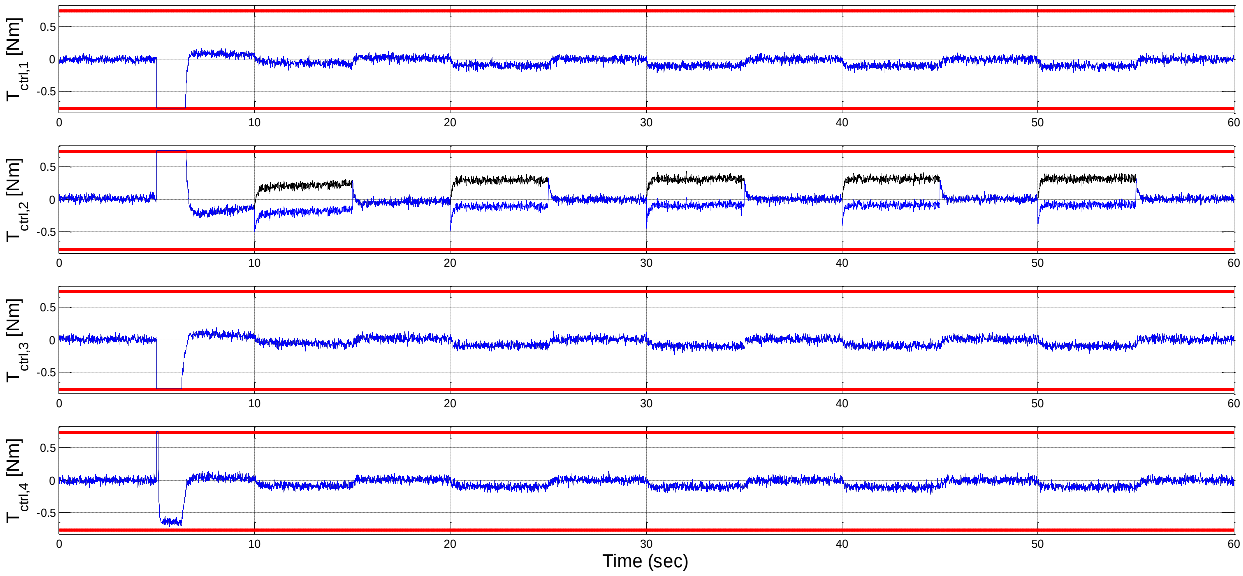


Figure 8.13 - Fault scenario n.3: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of rectangular pulse fault on the control input $T_{ctrl,2}$ and without fault accommodation.

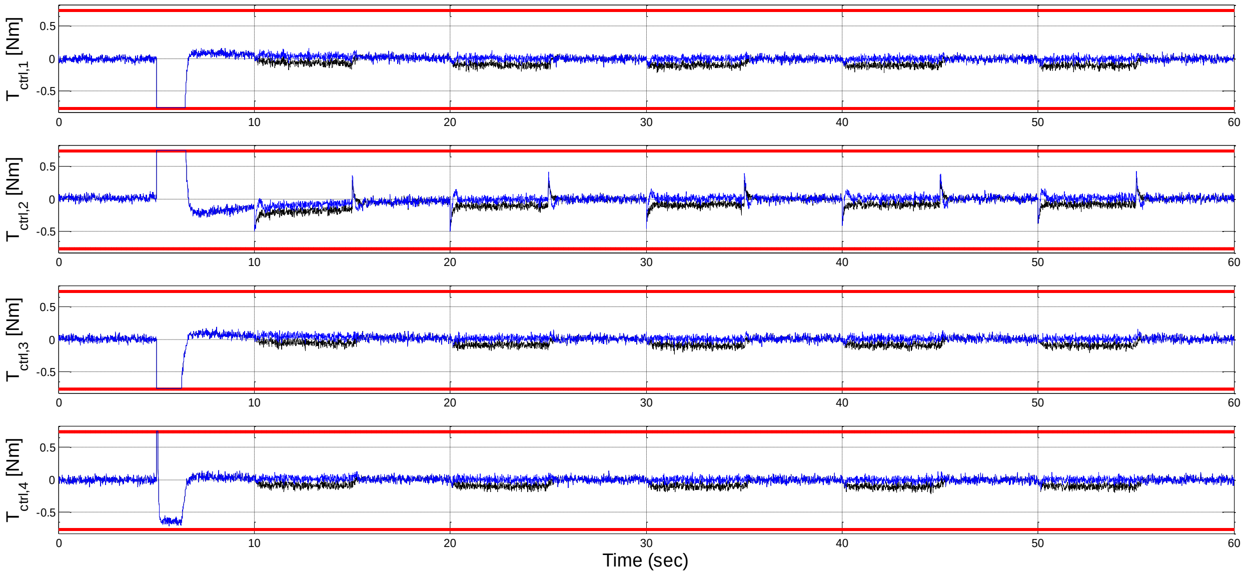


Figure 8.14 - Fault scenario n.3: faulty (black) and accommodated (blue) actuated control inputs T_{ctrl} in case of rectangular pulse fault on the control input $T_{ctrl,2}$ and with fault accommodation.

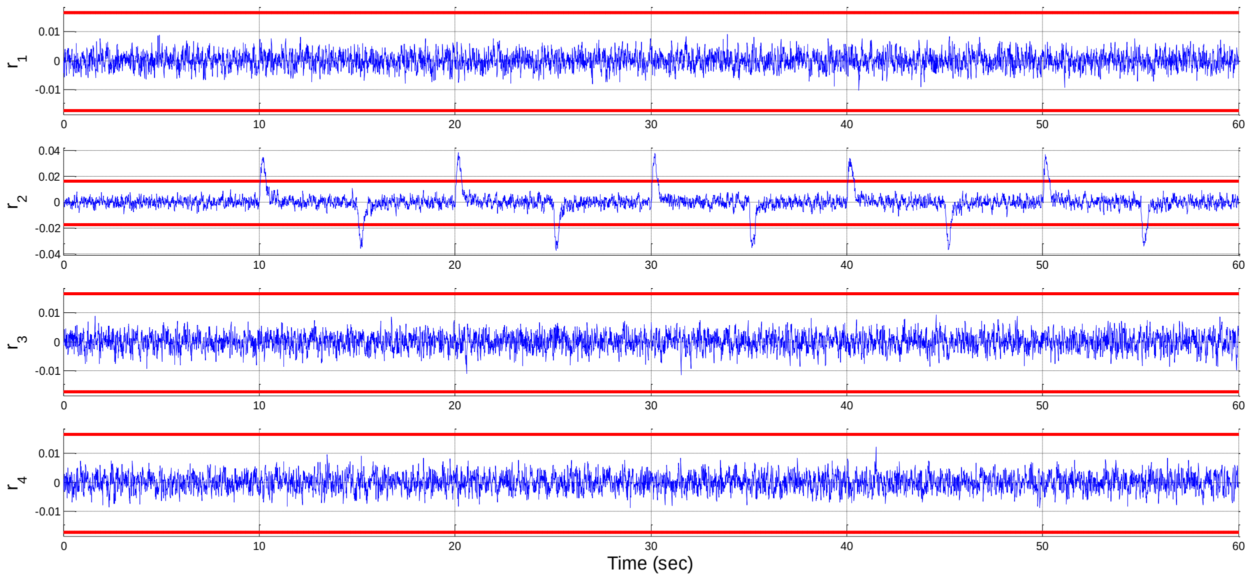


Figure 8.15 - Fault scenario n.3: local residuals of RB1 in case of rectangular pulse fault on the attitude control input $T_{ctrl,2}$ and subsequent fault accommodation.

Finally, Figs. 8.16, 8.17 and 8.18 show similar results in case of occurrence of a ramp fault affecting the second actuator. As previously stated, a longer period of time is required to pass before being able to detect and isolate the occurred incipient fault, but after the fault detection, correct isolation and accurate estimation the fault is compensated and the local residual of RB1 sensitive to the occurred fault returns to zero.

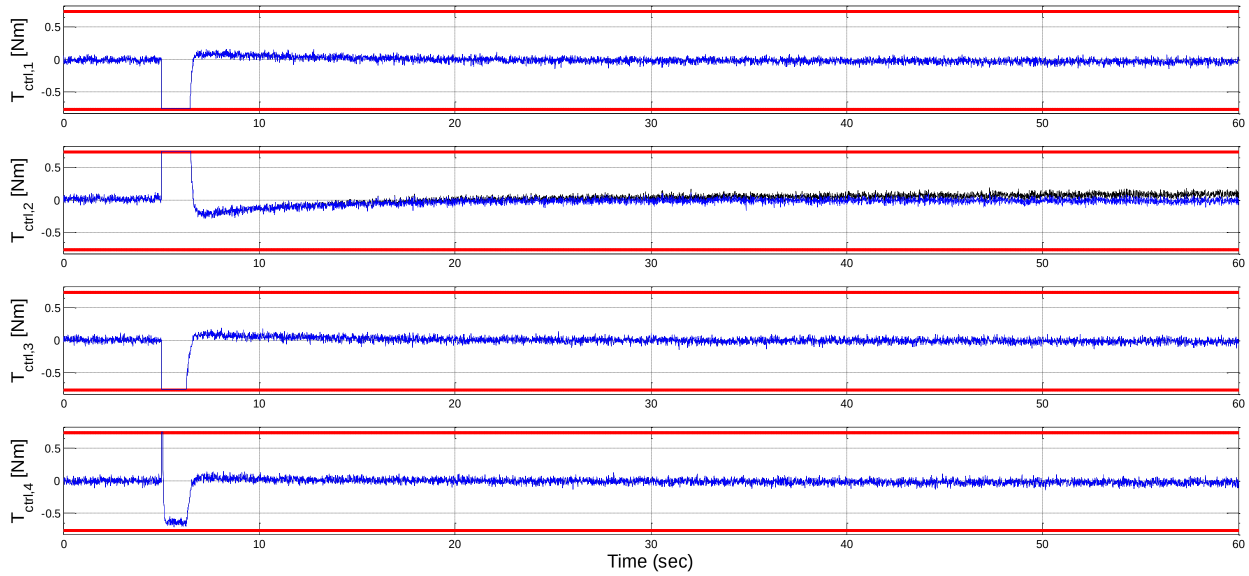


Figure 8.16 - Fault scenario n.4: commanded (black) and actuated (blue) control inputs T_c and T_{ctrl} in case of ramp fault on the control input $T_{ctrl,2}$ and without fault accommodation.

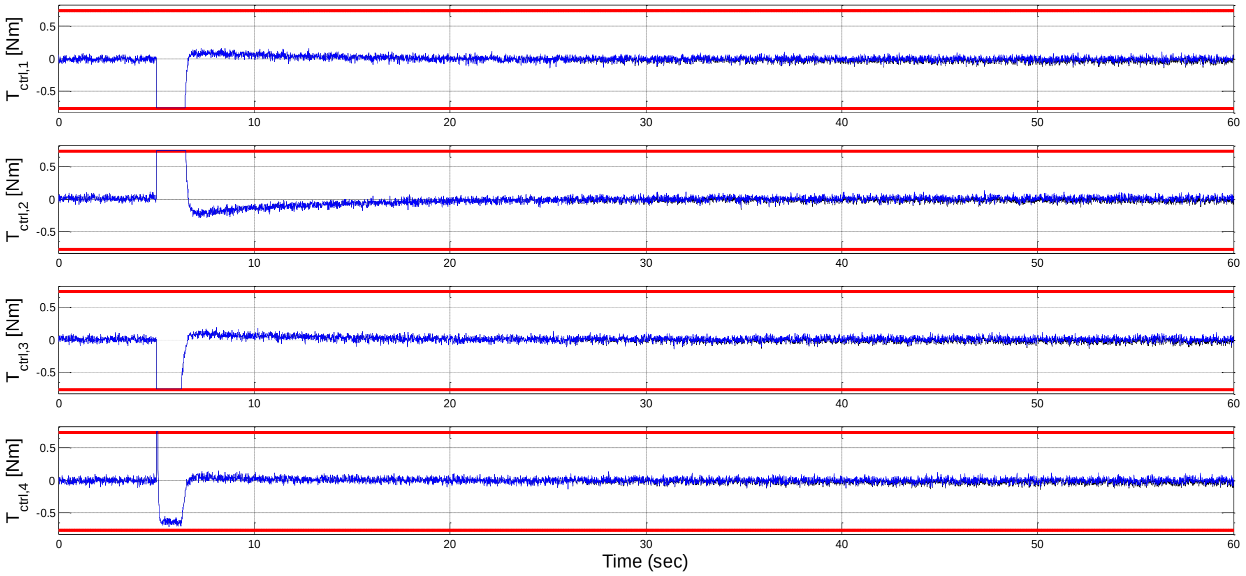


Figure 8.17 - Fault scenario n.4: faulty (black) and accommodated (blue) actuated control inputs T_{ctrl} in case of ramp fault on the control input $T_{ctrl,2}$ and with fault accommodation.

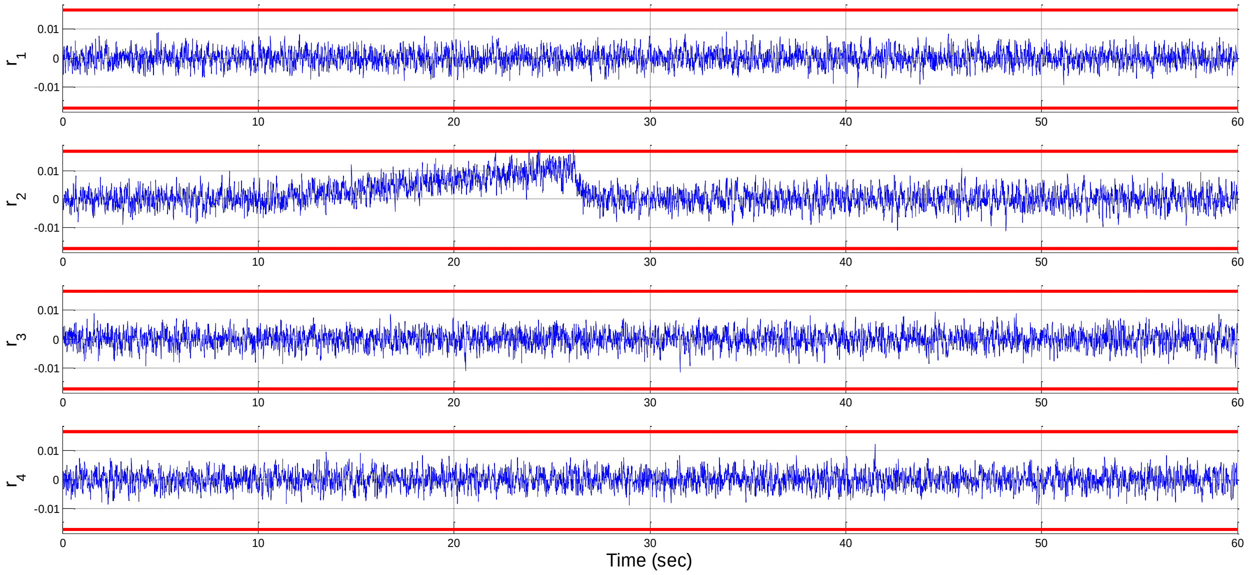


Figure 8.18 - Fault scenario n.4: local residuals of RB1 in case of ramp fault on the attitude control input $T_{ctrl,2}$ and subsequent fault accommodation.

On the other hand, considering the occurrence of faults affecting the flywheel spin rate sensors of the ACS, Figs. 8.19, 8.20 and 8.21 show the results obtained in case of step fault affecting the sensor measuring $\omega_{w,3}$. Figs. 8.19 and 8.20 show the comparison between the values of the actual unknown state variables and the corresponding measured flywheel spin rates without and with fault accommodation, respectively. As it can be seen, in Fig 8.19 it is evident the constant bias between the true and measured spin rate values after the fault occurrence, whereas Fig. 8.20 shows that the fault accommodation allows compensating this difference due to the fault presence. Fig. 8.21 shows that the local residual of RB1 sensitive to the occurred fault returns to zero after the fault accommodation. The global residuals of RB2 sensitive to the occurred fault present very similar behaviours, and they are omitted for the sake of simplicity. It is worth noting that, in case of step fault, the local residual of RB1 sensitive to the occurred fault returns to zero anyway, due to its sensitivity only to the fault input associated to the time derivative of the physical step fault, as shown in Fig. 4.11. However, in this case the transient phase is significantly shorted due to the prompt fault compensation once the fault has been detected and correctly isolated, as shown in Fig. 8.21.

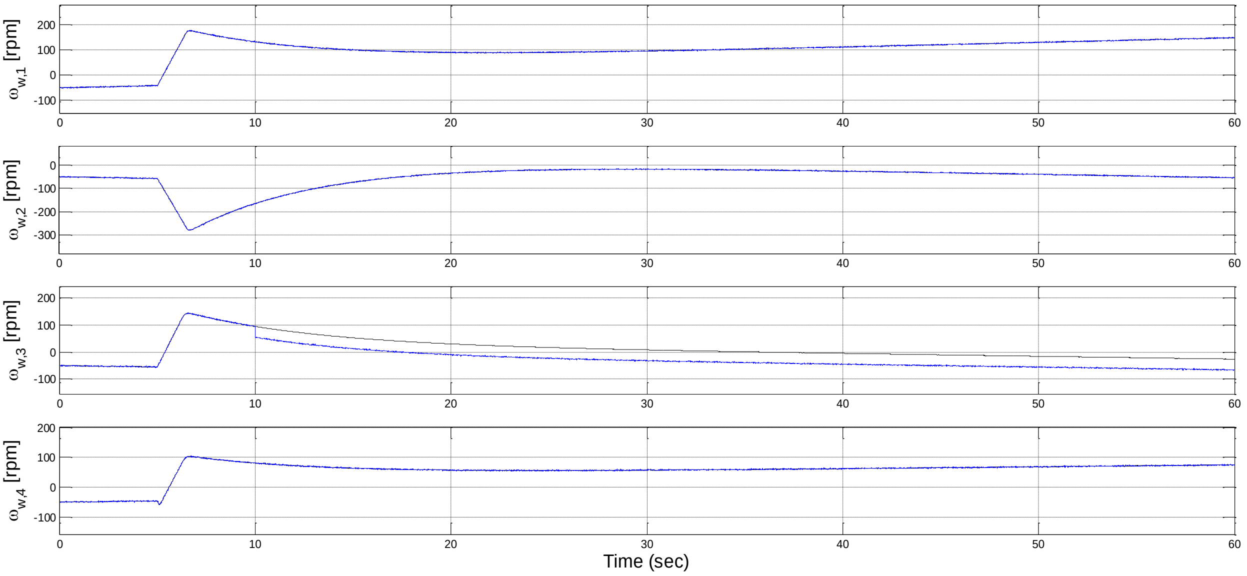


Figure 8.19 - Fault scenario n.6: true (black) and measured (blue) flywheel spin rates ω_w in case of step fault on the sensor output $\omega_{w,3_{measured}}$ and without fault accommodation.

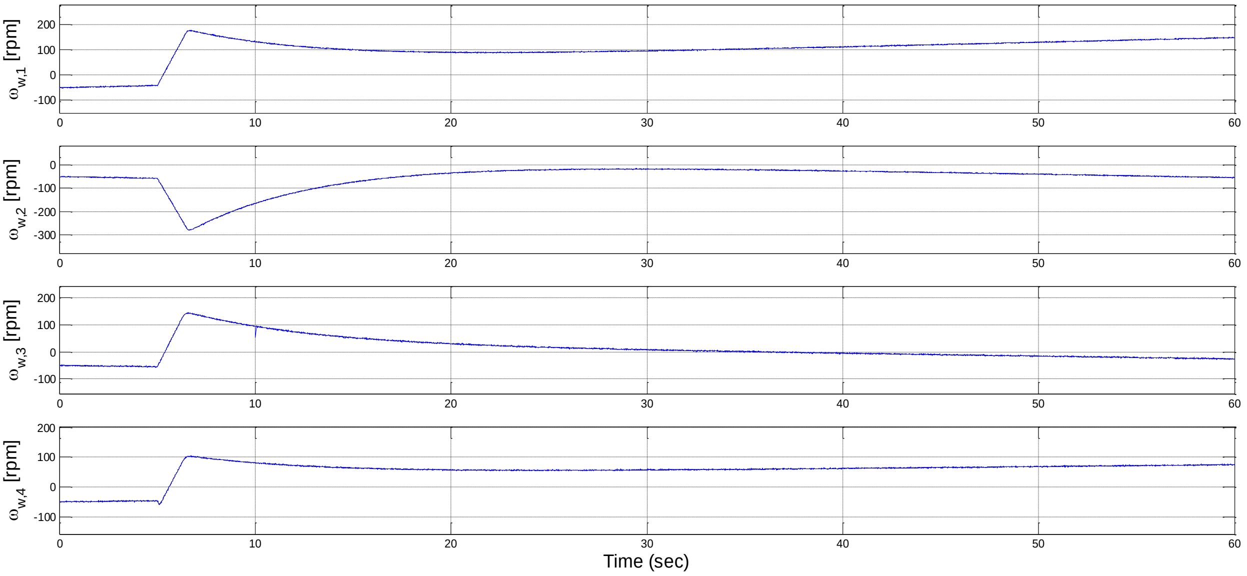


Figure 8.20 - Fault scenario n.6: true (black) and measured (blue) flywheel spin rates ω_w in case of step fault on the sensor output $\omega_{w,3_{measured}}$ and with fault accommodation.

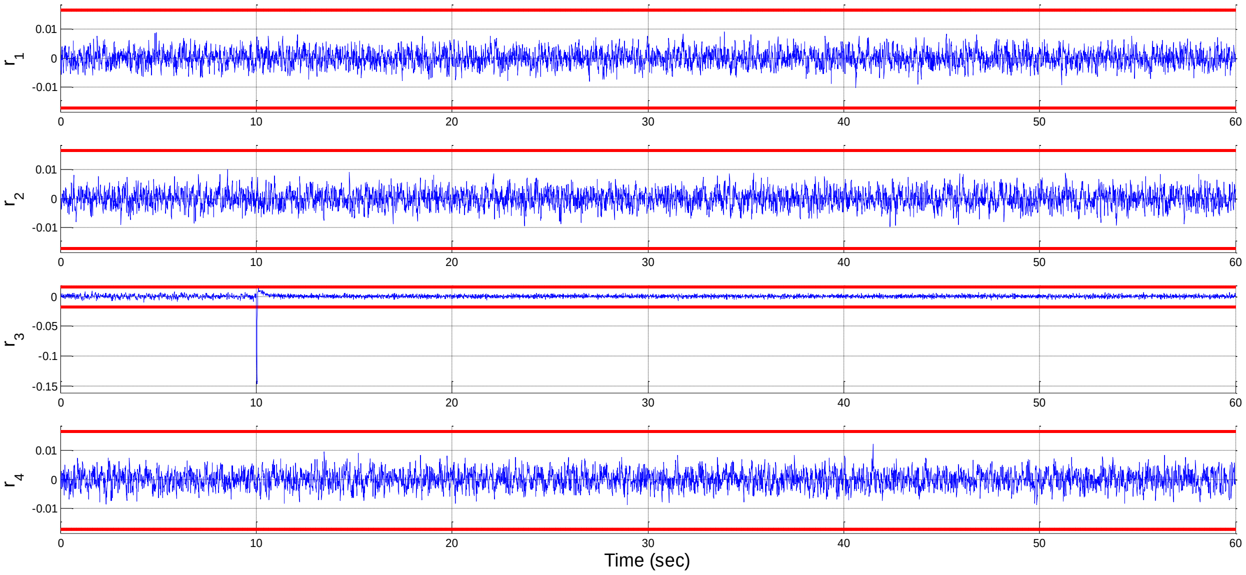


Figure 8.21 - Fault scenario n.6: local residuals of RB1 in case of step fault on the sensor output $\omega_{w,3_{measured}}$ and subsequent fault accommodation.

Figs. 8.22, 8.23 and 8.24 show analogous results obtained in case of a sinusoidal fault affecting the sensor measuring $\omega_{w,3}$. In case of fault accommodation, the effects of the occurred fault on the sensor measurement are almost completely compensated, as shown in Fig. 8.23. A small oscillation of the sensitive local residual of RB1 lasts in Fig. 8.24, because of the remaining fault effects due to the not null (but bounded) fault estimation error shown in Fig. 6.21 and the subsequent not perfect fault compensation. Anyway, the fault accommodation makes the residual sensitive to the occurred fault to remain bounded within small values, in comparison to the case without fault accommodation shown in Fig. 4.13.

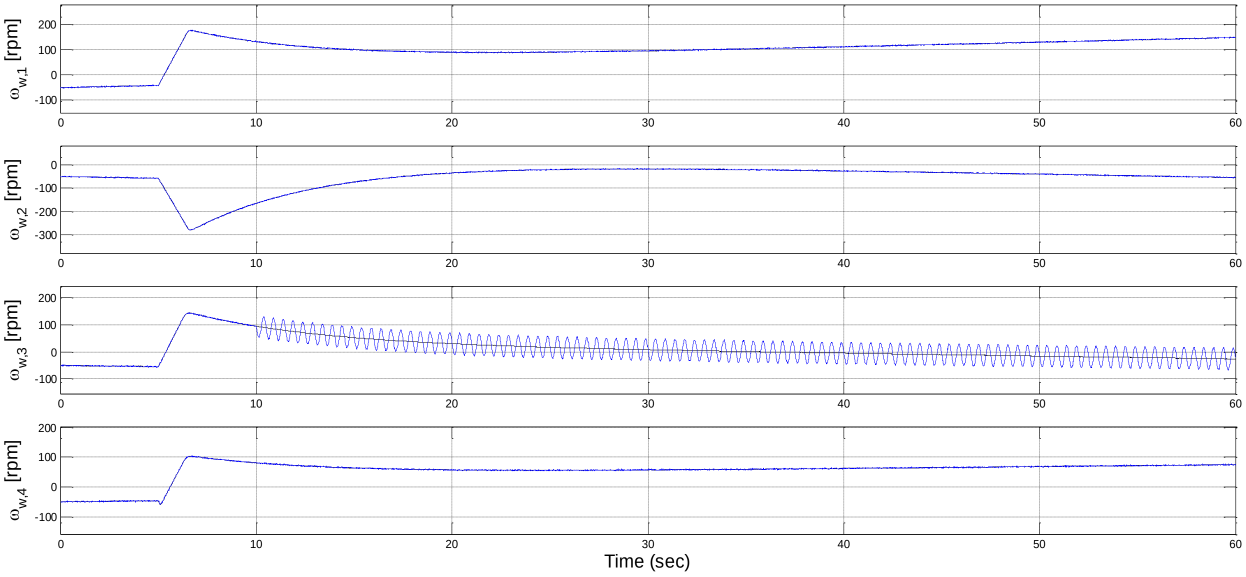


Figure 8.22 - Fault scenario n.7: true (black) and measured (blue) flywheel spin rates ω_w in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$ and without fault accommodation.

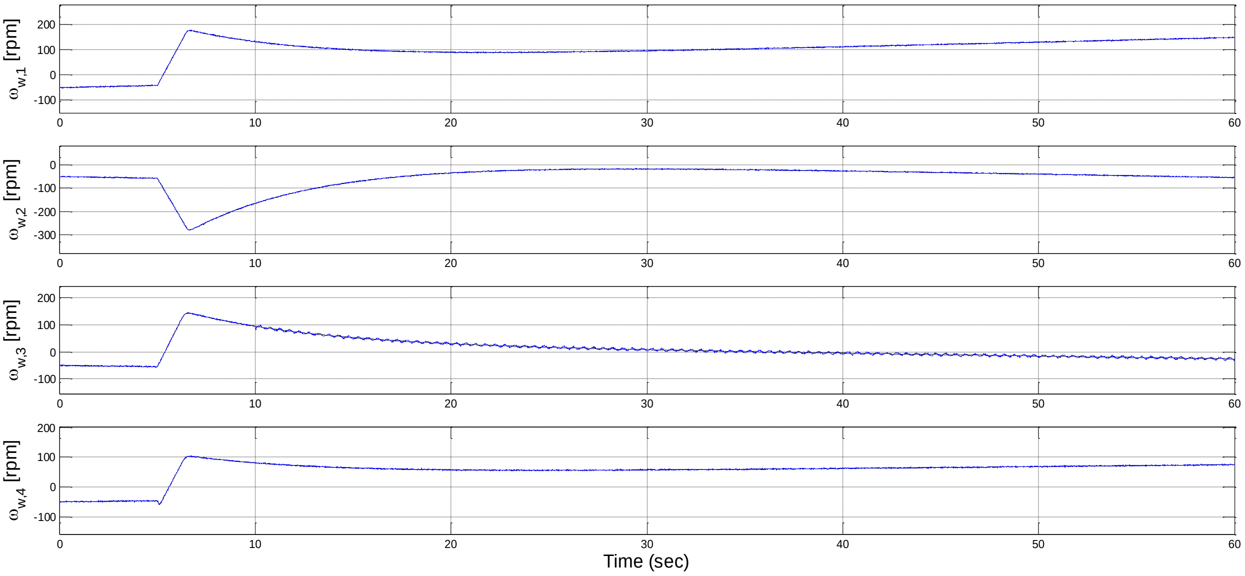


Figure 8.23 - Fault scenario n.7: true (black) and measured (blue) flywheel spin rates ω_w in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$ and with fault accommodation.

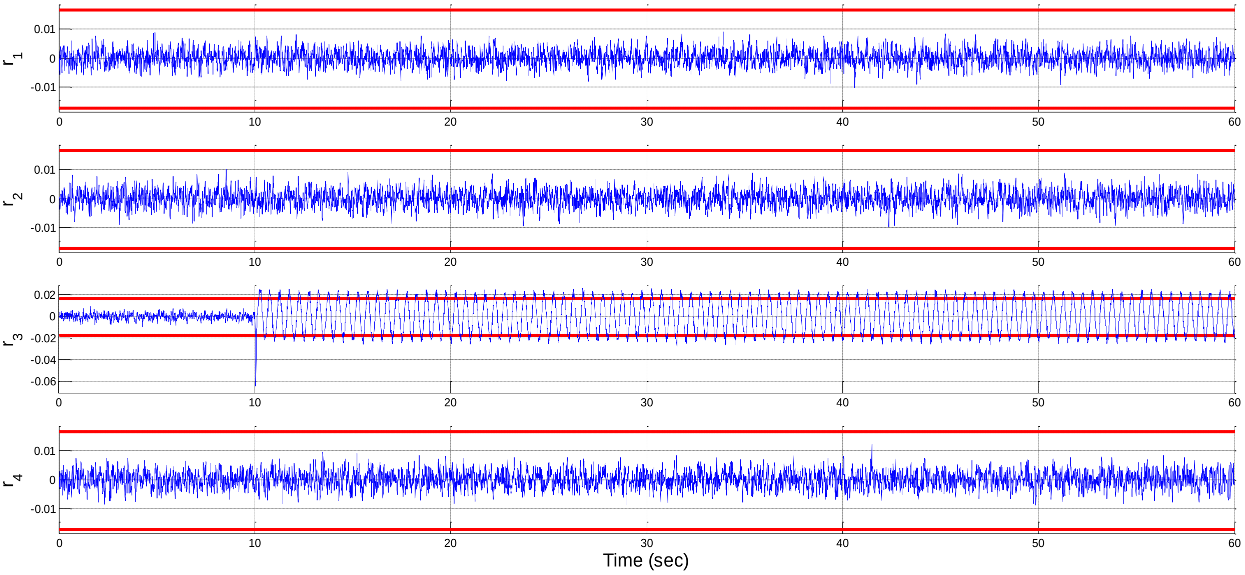


Figure 8.24 - Fault scenario n.7: local residuals of RB1 in case of sinusoidal fault on the sensor output $\omega_{w,3_{measured}}$ and subsequent fault accommodation.

Figs. 8.25, 8.26 and 8.27 show results relatively similar to the previous cases, but obtained in case of failure of the sensor measuring $\omega_{w,3}$. In this case, the estimated fault results to correspond to the actual value of the state variable to be measured. Therefore, this estimation, and hence the corresponding accommodated sensor output, can be considered as the output of a software sensor.

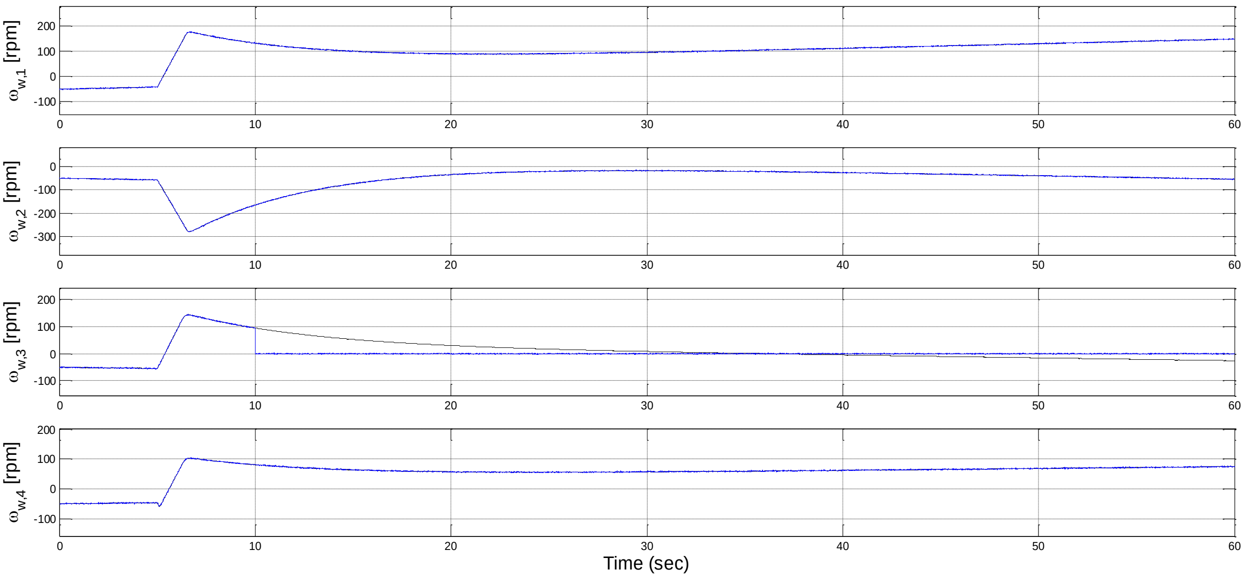


Figure 8.25 - Fault scenario n.8: true (black) and measured (blue) flywheel spin rates ω_w in case of failure of the sensor providing the measurement of $\omega_{w,3}$ and without fault accommodation.

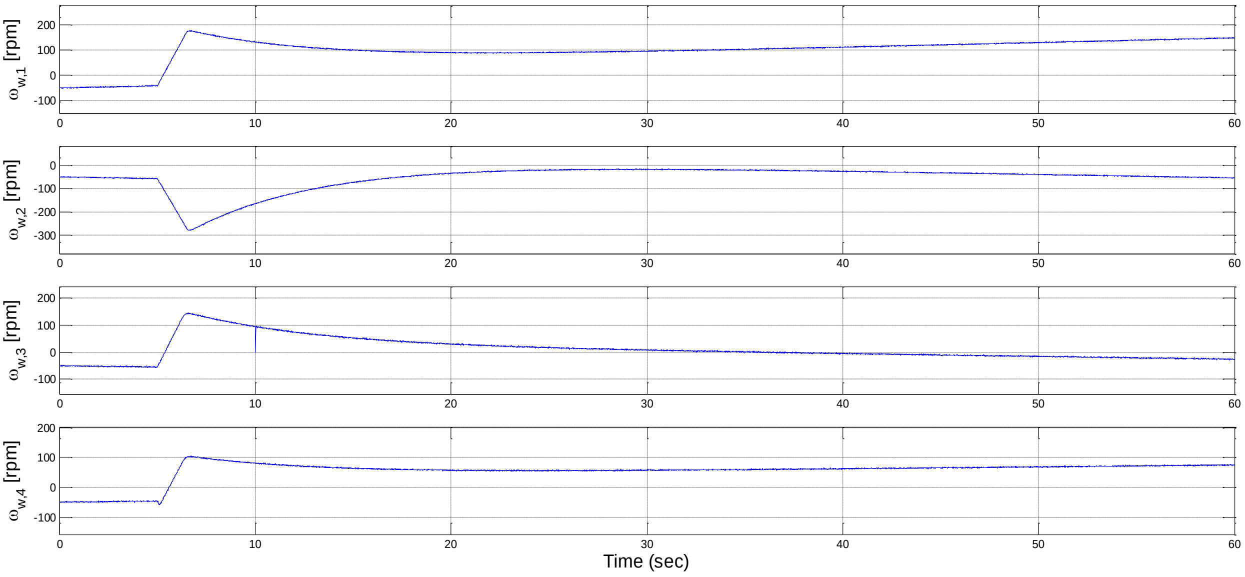


Figure 8.26 - Fault scenario n.8: true (black) and measured (blue) flywheel spin rates ω_w in case of failure of the sensor providing the measurement of $\omega_{w,3}$ and with fault accommodation.

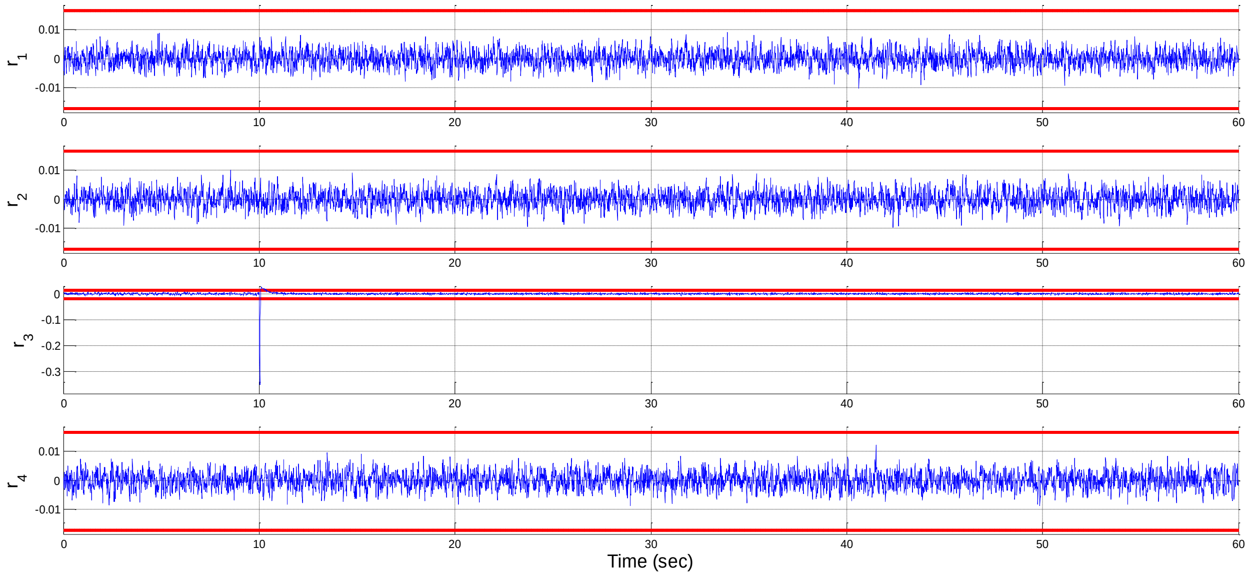


Figure 8.27 - Fault scenario n.8: local residuals of RB1 in case of failure of the sensor providing the measurement of $\omega_{w,3}$ and subsequent fault accommodation.

8.2.3 Fault Accommodation after Sensor Faults in the Satellite ADS

Now, considering the occurrence of faults affecting the sensors of the satellite ADS, Figs. 8.28, 8.29, 8.30 and 8.31 show the results obtained in case of step fault affecting the sensor measuring $\omega_{x,i}$. Figs. 8.28 and 8.29 show the comparison between the values of the actual unknown state variables and the corresponding measured satellite angular velocities without and with fault accommodation, respectively. Again, in Fig 8.28 it is evident the constant bias between the true and measured angular velocity values after the fault occurrence, whereas Fig. 8.29 shows that the fault accommodation allows compensating this difference due to the fault presence.

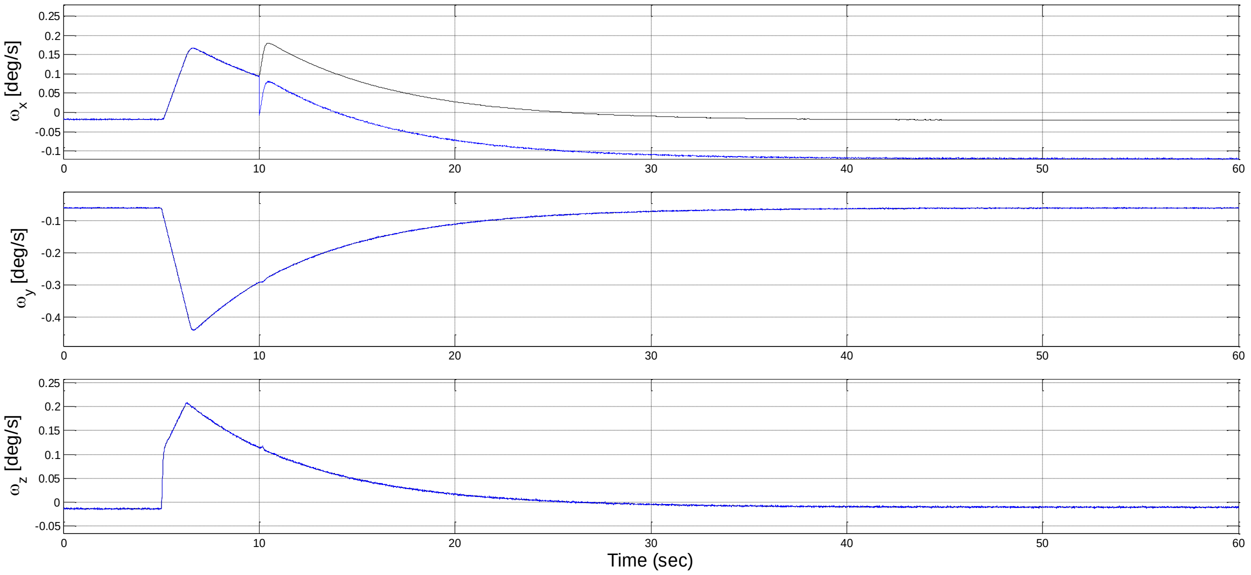


Figure 8.28 - Fault scenario n.9: true (black) and measured (blue) satellite angular velocity ω in case of step fault on the sensor output $\omega_{x,i_{measured}}$ and without fault accommodation.

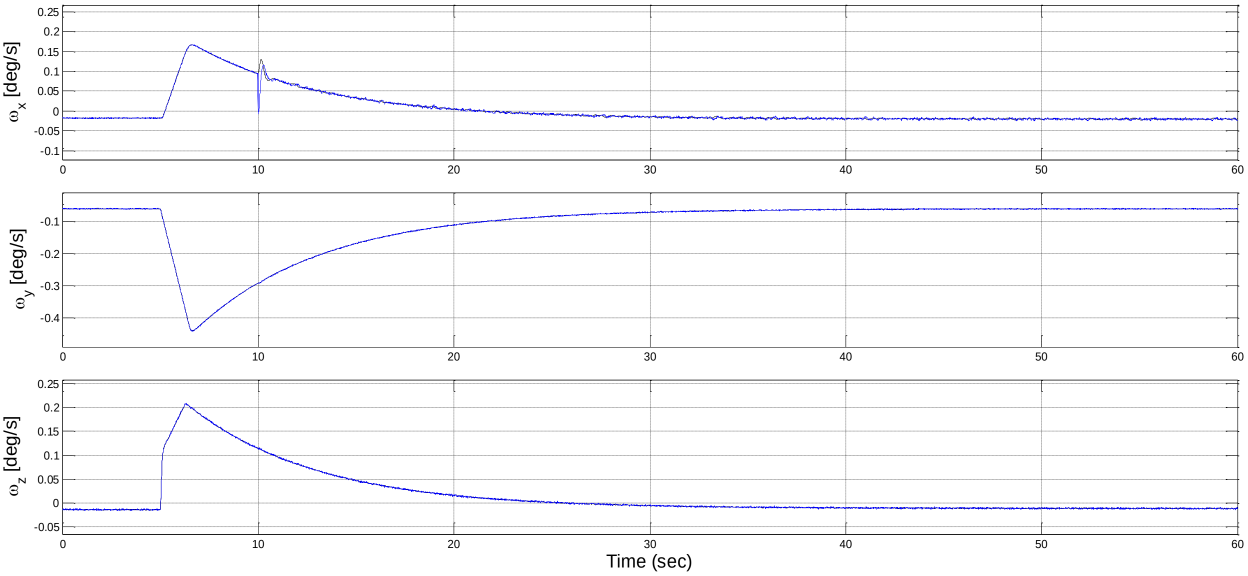


Figure 8.29 - Fault scenario n.9: true (black) and measured (blue) satellite angular velocity ω in case of step fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

Figs. 8.30 and 8.31 show that the residuals of RB3 sensitive to the occurred fault returns to zero after the fault accommodation. The corresponding residuals of RB4 sensitive to the same occurred fault present identical behaviours, and they are omitted for the sake of simplicity.

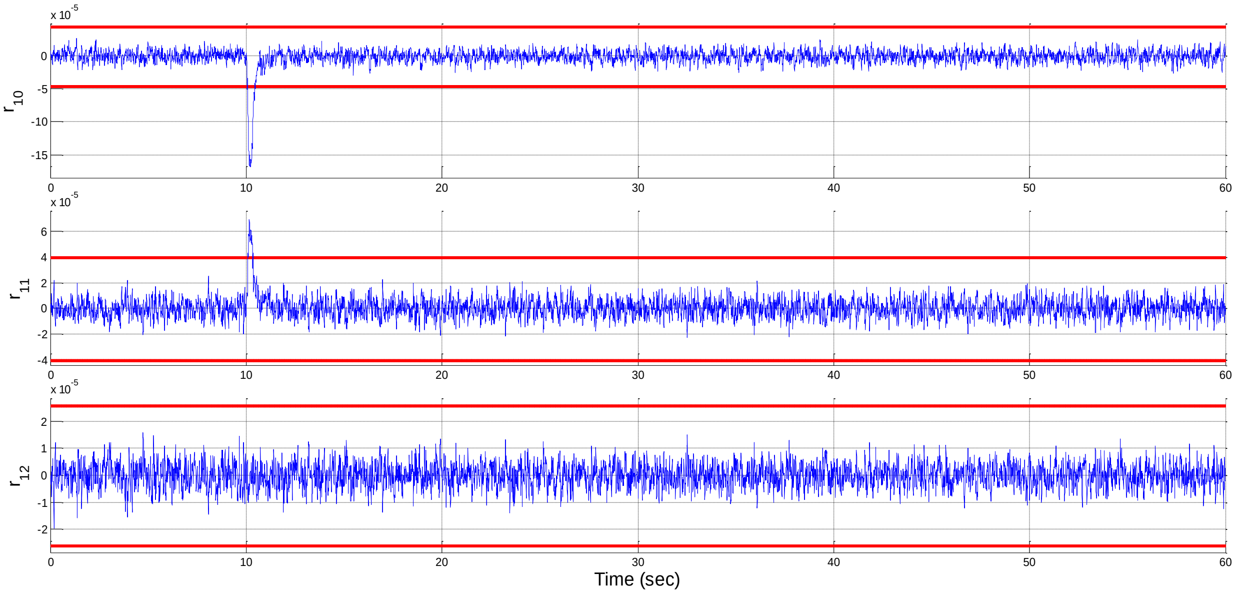


Figure 8.30 - Fault scenario n.9: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$ in case of step fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

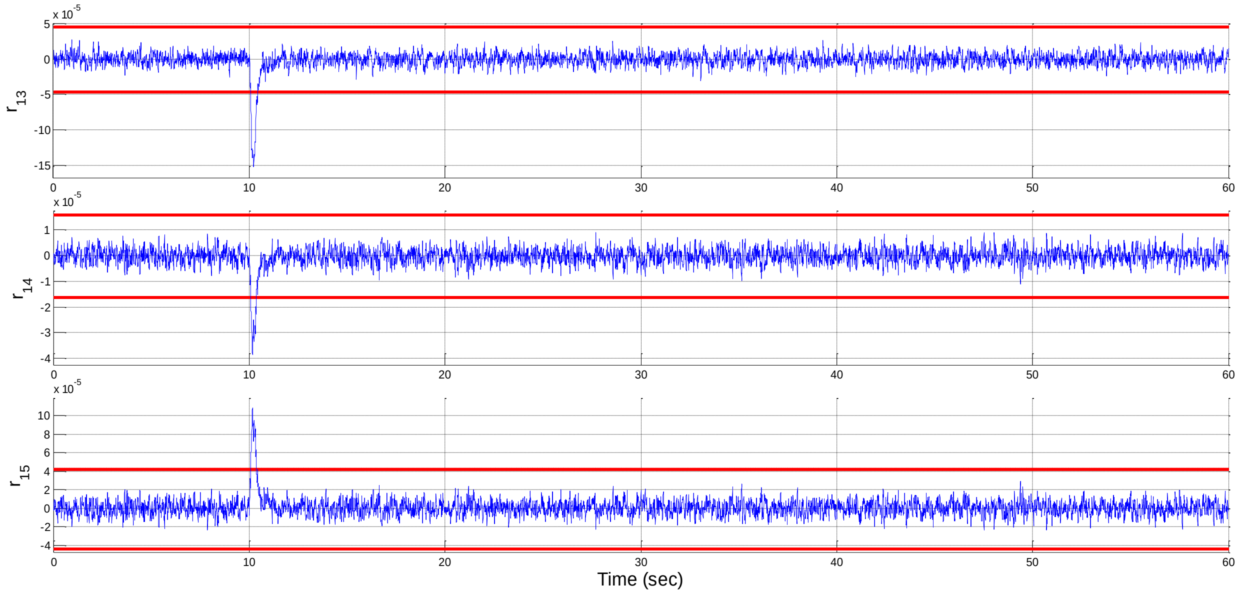


Figure 8.31 - Fault scenario n.9: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$ in case of step fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

Figs. 8.32, 8.33, 8.34 and 8.35 show analogous results obtained in case of a lock-in-place fault affecting the sensor measuring $\omega_{x,i}$. In case of fault accommodation, the effects of the occurred fault on the sensor measurement are almost completely compensated, as shown in Fig. 8.33. Oscillations of the sensitive residuals of RB3 last in Figs. 8.34 and 8.35, because of the remaining fault effects due to the not null (but bounded) fault estimation error shown in Fig. 6.30 and the subsequent not perfect fault compensation. Anyway, the fault accommodation makes the residuals sensitive to the occurred fault to remain bounded between small values, in comparison to the case without fault accommodation shown in Figs. 4.23 and 4.24.

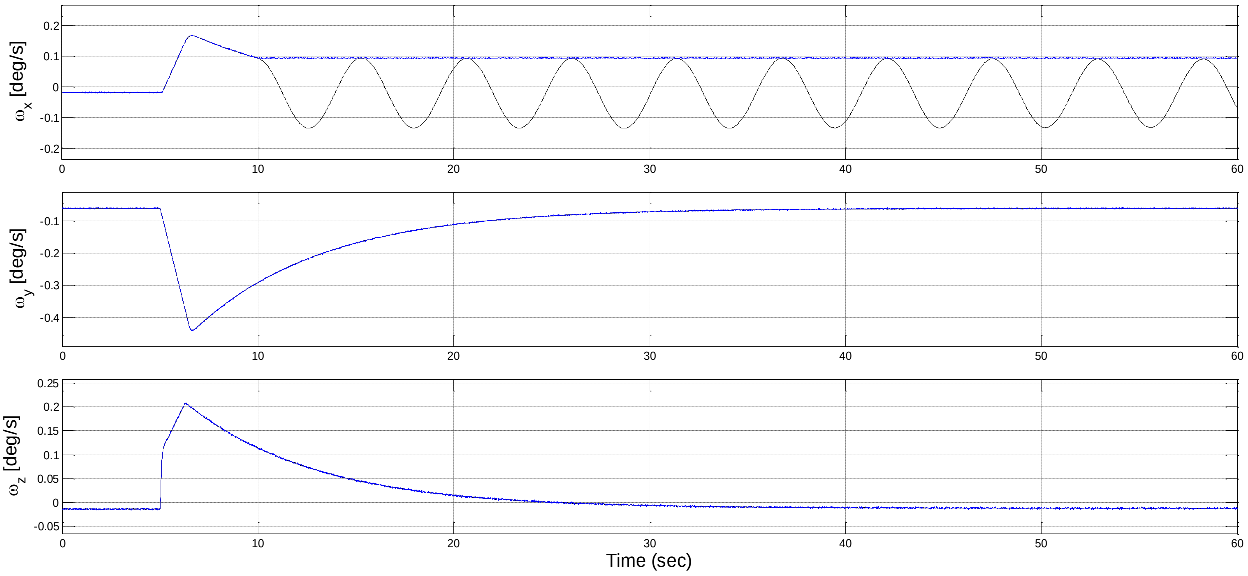


Figure 8.32 - Fault scenario n.10: true (black) and measured (blue) satellite angular velocity ω in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$ and without fault accommodation.

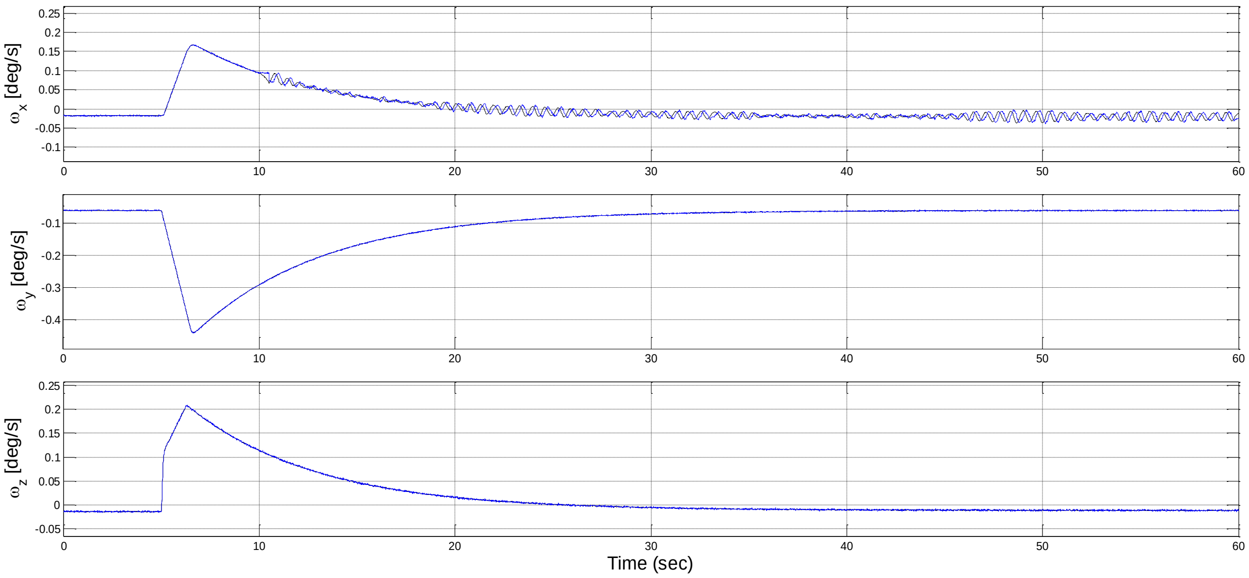


Figure 8.33 - Fault scenario n.10: true (black) and measured (blue) satellite angular velocity ω in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

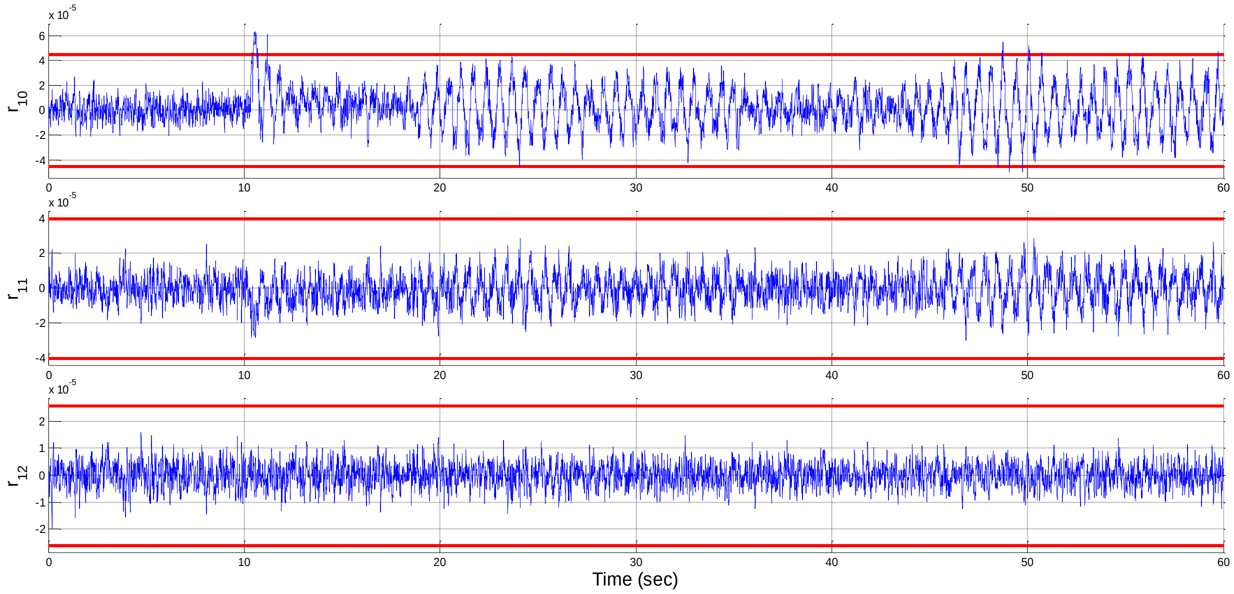


Figure 8.34 - Fault scenario n.10: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$ in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

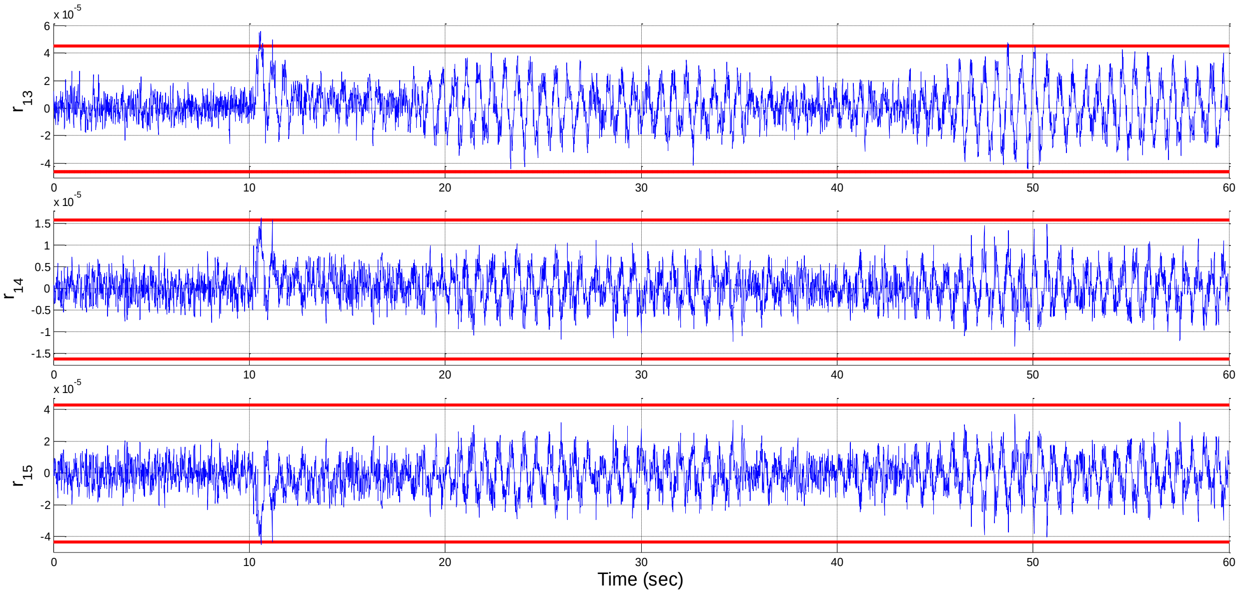


Figure 8.35 - Fault scenario n.10: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$ in case of lock-in-place fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

Again, Figs. 8.36, 8.37, 8.38 and 8.39 show analogous results obtained in case of a loss-of-effectiveness fault affecting the sensor measuring $\omega_{x,i}$.

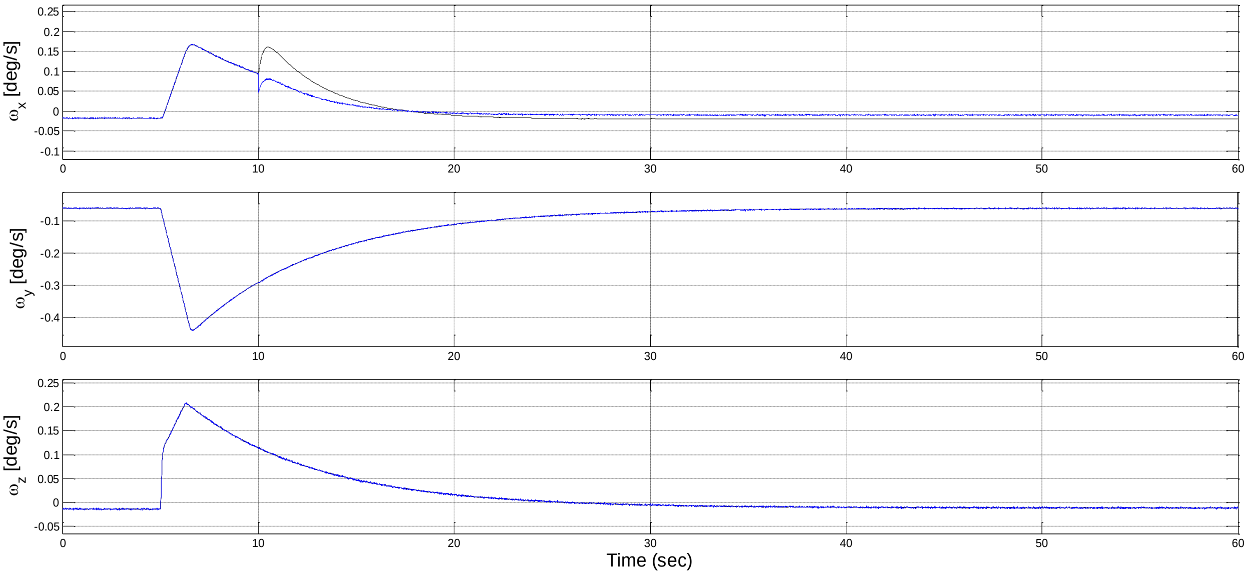


Figure 8.36 - Fault scenario n.11: true (black) and measured (blue) satellite angular velocity ω in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$ and without fault accommodation.

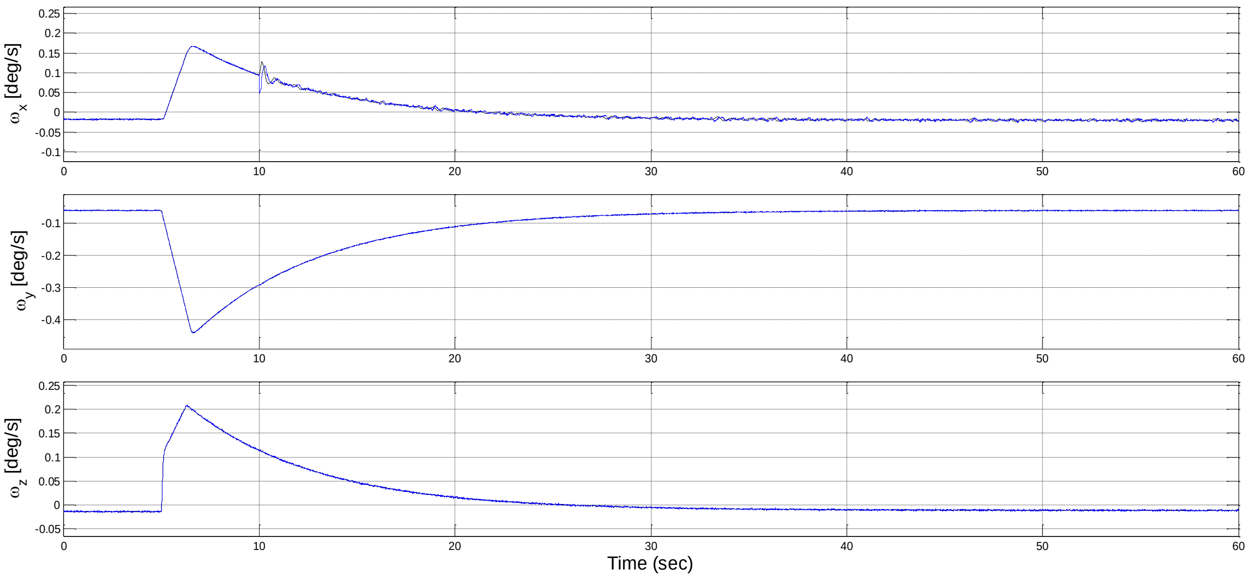


Figure 8.37 - Fault scenario n.11: true (black) and measured (blue) satellite angular velocity ω in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

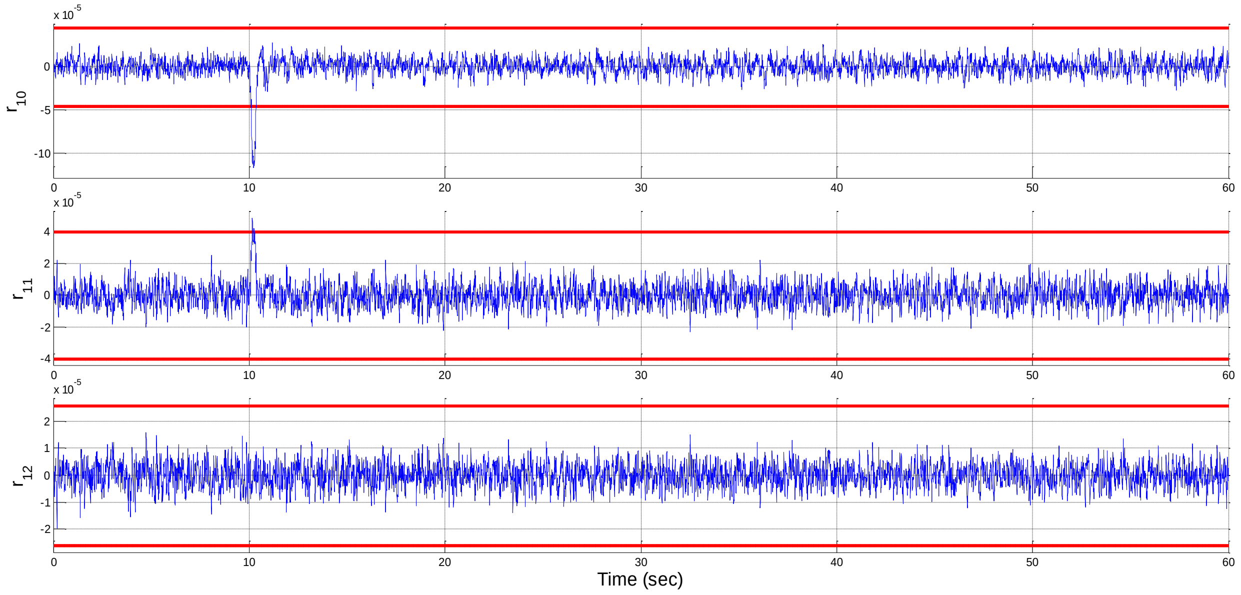


Figure 8.38 - Fault scenario n.11: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{y,i}$ in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

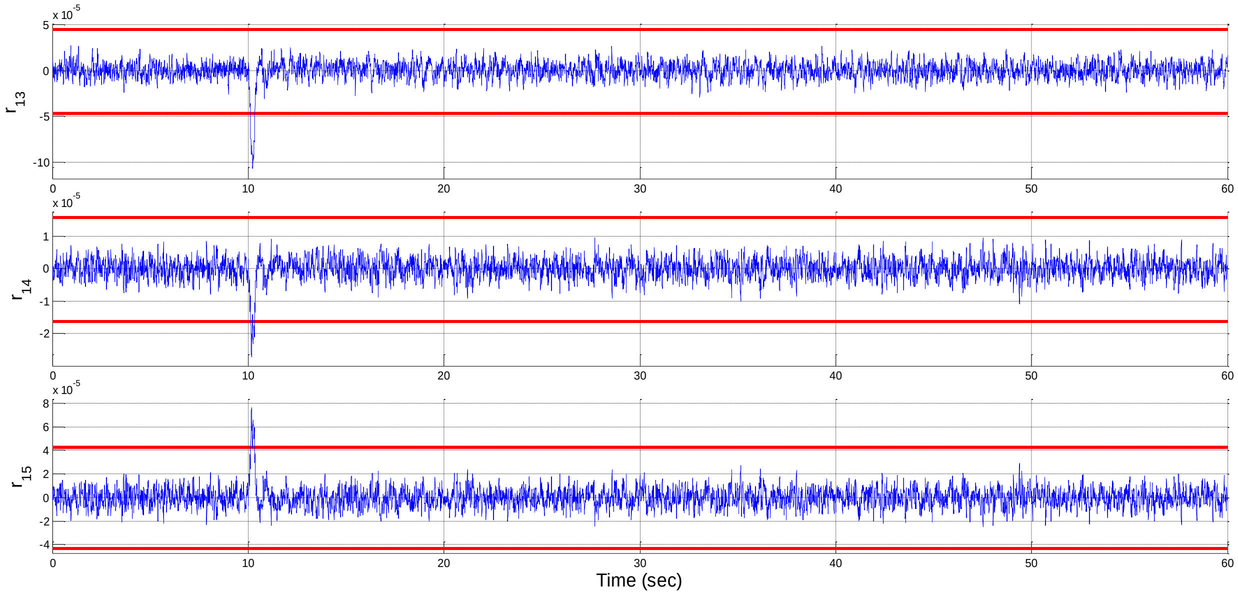


Figure 8.39 - Fault scenario n.11: residuals of RB3, sensitive to fault inputs associated to the physical faults affecting the sensors measuring $\omega_{x,i}$ and $\omega_{z,i}$ in case of loss-of-effectiveness fault on the sensor output $\omega_{x,i_{measured}}$ and with fault accommodation.

Finally, Figs. 8.40, 8.41, 8.42 and 8.43 show analogous results obtained in case of an additive fault $\overline{\mathbf{F}}_{\overline{\mathbf{q}}_{star}}$ affecting the sensor measuring the attitude quaternion vector $\overline{\mathbf{q}}_{star1}$. In this case, all the residuals of RB3 are in general sensitive to the occurred fault. Again, all the residuals of RB3 presenting deviations from their fault-free conditions due to the fault occurrence quickly return to zero after the fault accommodation.

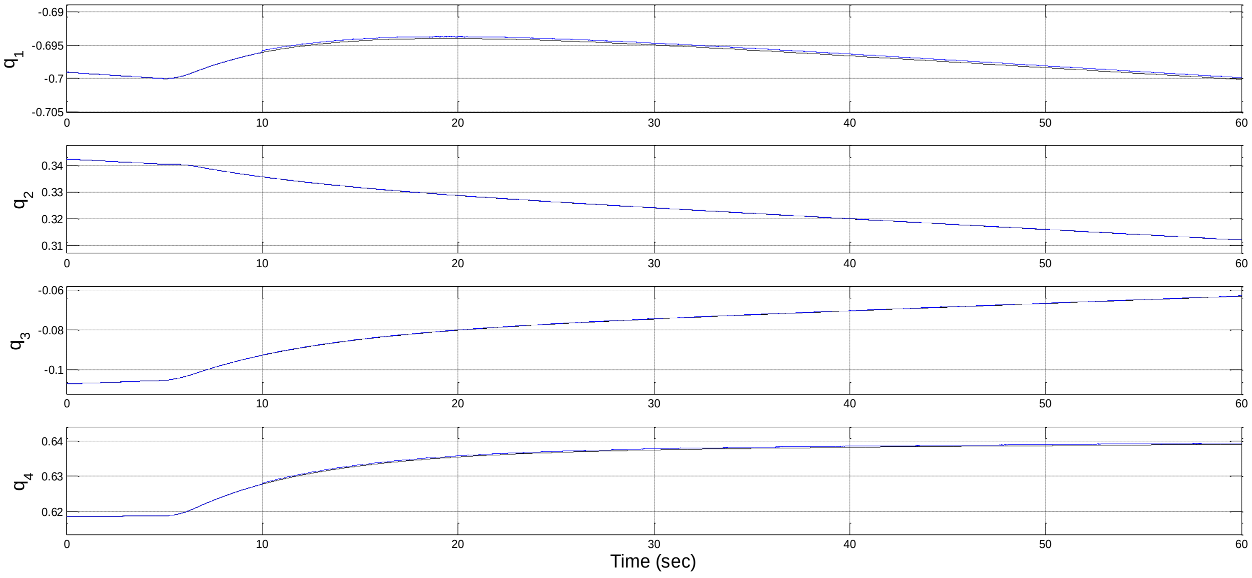


Figure 8.40 - Fault scenario n.12: true (black) and measured (blue) attitude quaternion vector $\overline{\mathbf{q}}_{star1}$ in case of fault on the sensor output and without fault accommodation.

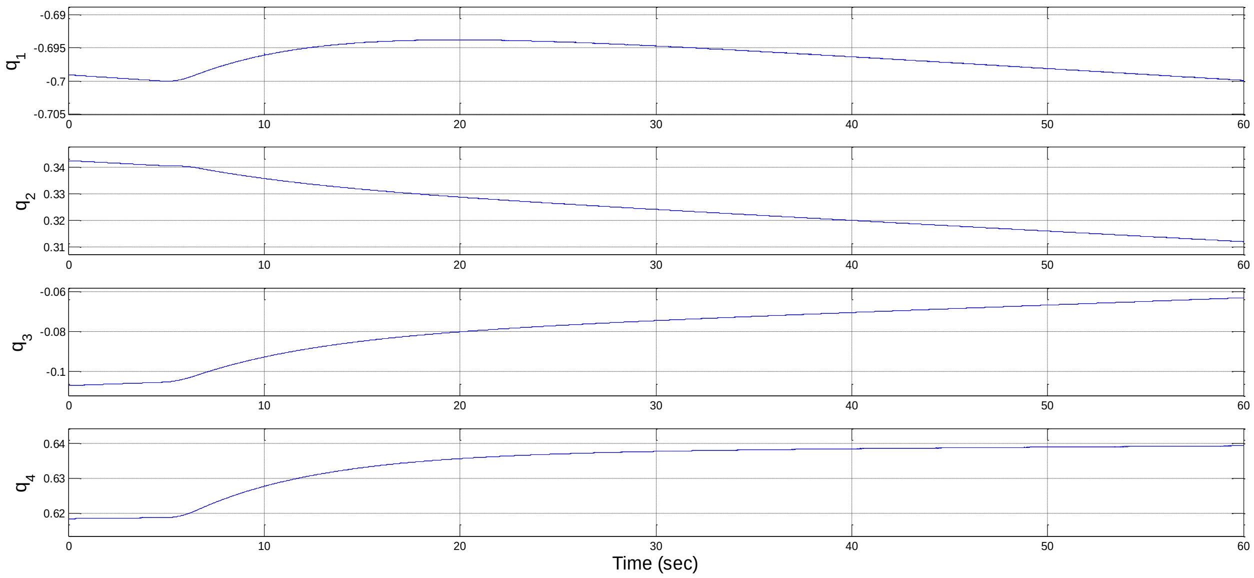


Figure 8.41 - Fault scenario n.12: true (black) and measured (blue) attitude quaternion vector $\overline{\mathbf{q}}_{star1}$ in case of fault on the sensor output and with fault accommodation.

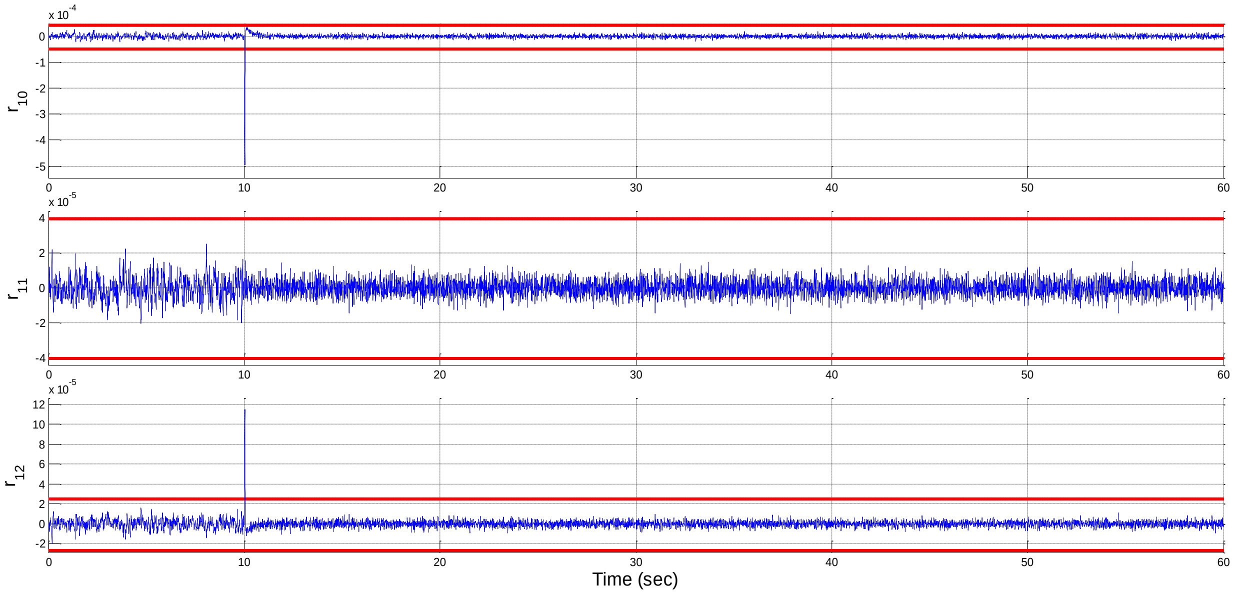


Figure 8.42 - Fault scenario n.12: first triad of residuals of RB3 in case of fault on the sensor measuring $\overline{\mathbf{q}}_{star1}$ and subsequent fault accommodation.

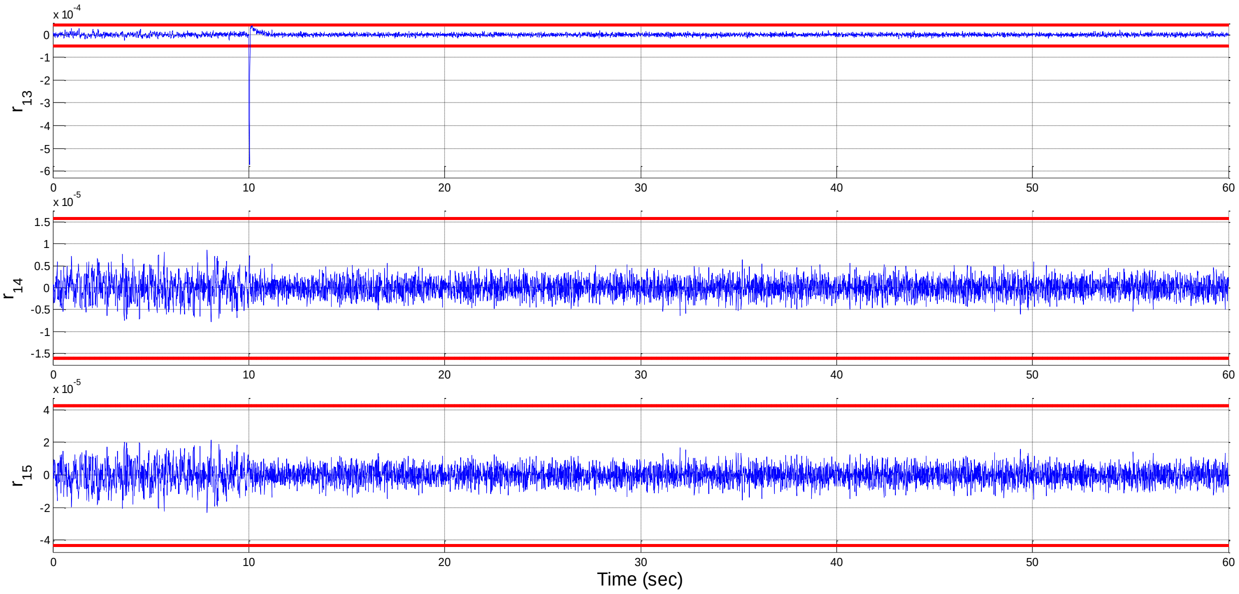


Figure 8.43 - Fault scenario n.12: second triad of residuals of RB3 in case of fault on the sensor measuring $\overline{\mathbf{q}}_{star1}$ and subsequent fault accommodation.

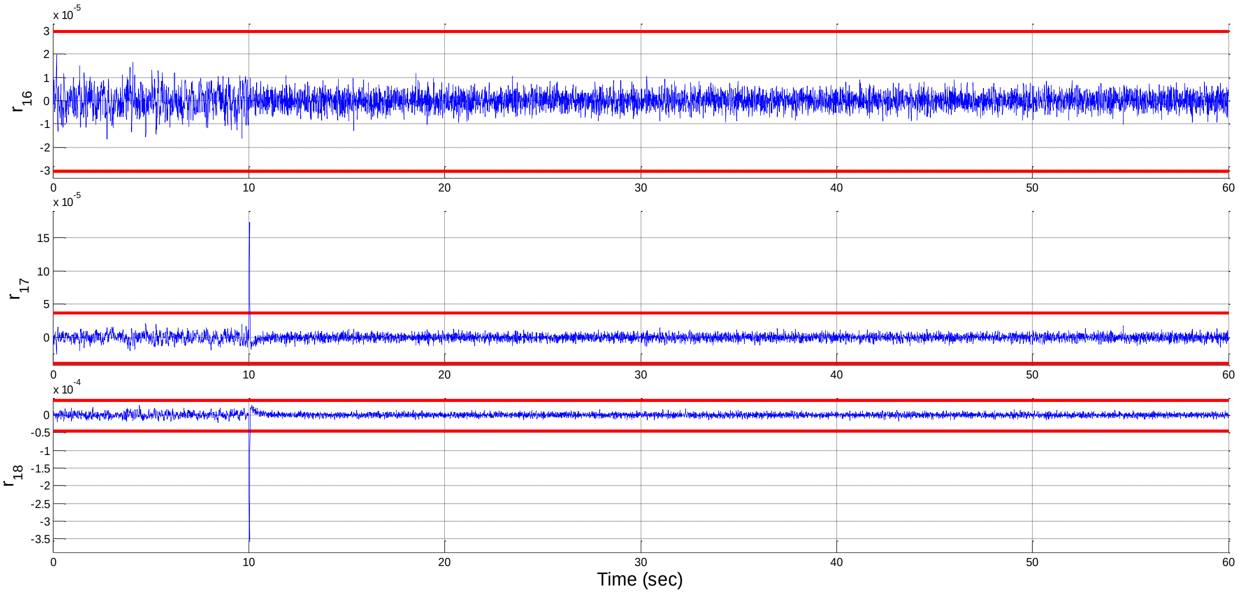


Figure 8.44 - Fault scenario n.12: last triad of residuals of RB3 in case of fault on the sensor measuring $\overline{\mathbf{q}}_{star1}$ and subsequent fault accommodation.

9 CONCLUSION

Modern control systems are becoming more and more complex and control algorithms more and more sophisticated. Consequently, fault detection and diagnosis and fault tolerant control have gained central importance over the past decades, due to the increasing requirements of availability, cost efficiency, reliability and operating safety.

In the context of aerospace applications, this thesis dealt with the FDI, FDD and FTC problems for an attitude control application of a spacecraft, in particular focusing on fault tolerant control and diagnosis of possible faults affecting the actuators and sensors of an attitude determination and control system of a LEO satellite. This dissertation aimed to summarize and extend the results previously presented by the author, and extend the problem of fault diagnosis also to sensor faults, in addition to actuator faults only.

A novel fault diagnosis and fault tolerant control scheme has been developed for the detection, isolation, estimation and accommodation of possible faults affecting the control torques provided by reaction wheel actuators, the flywheel spin rate measurements, attitude and angular velocity measurements provided by the sensors of the satellite ADCS. The proposed diagnosis scheme is based on the exploitation of a FDD system, which is composed of a FDI module and a FE module.

The designed FDI module is composed of a set of scalar model-based residual generators, which have been organized in four independent banks of filters working in parallel, each of them specifically dedicated to the detection and isolation of possible faults affecting specific sets of components of the ADCS.

The exploitation of such banks of residual filters allowed to realize a complete procedure for fault detection and isolation based on the exploitation only of sensor measurements that are intended and exploited for attitude control purpose, and in general always available to the diagnosis system.

In this way, it is worth observing that the internal electrical models of actuators and sensors have been neglected in this thesis, *i.e.* no fault diagnosis has been performed using local electrical measurements of current or voltage or other types of direct internal checks in the supervised system components. This would permit to reduce the complexity of the fault diagnosis system thanks to the exploitation of sensors, which are always present in every satellite, avoiding the need for further integrated supervision and monitoring systems.

Due to the aerodynamic disturbance uncertainty in LEO satellites, an explicit disturbance decoupling approach based on the NLGA allowed to design residual filters not affected by the aerodynamic parametric uncertainty. Moreover, a general procedure for modelling of the sensor faults in order to obtain a new spacecraft nonlinear dynamic model, which is affine with respect to both the actuator and sensor faults and suitable to the application of the NLGA has been exploited.

The overall detection and isolation procedure for the considered actuator and sensor faults is carried out by exploiting a cross-check of all the generated residuals, by means of a proper decision logic and residual comparison scheme. In this context, the joint use the four banks of residual filters in the FDI system allowed riding over the limitations of each bank of residual filters for the detection and isolation of faults in the supervised subsystems.

The overall FDD system have been completed by a FE module, which consists of a bank of adaptive observers exploited to obtain accurate and quick fault estimates. This bank of adaptive observers is based on RBF-NNs. The on-line learning capability of the Radial Basis Function Neural Network allowed obtaining accurate adaptive estimates of the occurred faults. Moreover, the use of a Radial Basis Function Neural Network allowed designing generalized fault estimation adaptive observers, which do not need any a priori information about the fault internal model. The outputs of the fault estimation module are enabled once a fault has been correctly detected and isolated by the previously designed FDI module.

Finally, an AFTC system has been realized by implementing a fault tolerant strategy, based on the information from the FDI/FDD system. In case of actuator faults, a fault accommodation scheme has been exploited for soft faults, when the faulty actuator is yet operative but with a degraded performance, by exploiting the fault estimation information from the FE module. On the contrary, a control reconfiguration scheme is used in case of hard actuator faults by excluding the effects of the faulted actuator in the system by exploiting the actuator redundancy and using the control inputs of the other actuators appropriately to maintain the complete attitude controllability. In this way, a FDIR scheme has been realized, without directly exploiting any fault estimation information from the FE module. In case of sensor faults, a fault accommodation scheme is used by directly exploiting the fault estimation information from the FE module.

The performances of the proposed fault diagnosis and fault tolerant control strategies have been evaluated when applied to a detailed nonlinear spacecraft attitude model taking account also of measurement noise, and exogenous disturbance signals. In particular, these exogenous disturbance terms are represented by aerodynamic and gravitational disturbances.

However, as the gravitational disturbance model is almost perfectly known, the FDI robustness has been achieved by exploiting an explicit disturbance decoupling method, based on the NLGA, applied only to the aerodynamic force term. This term represents the main source of uncertainty in the satellite dynamic model, mainly due to the lack of knowledge of the accurate values of air density and satellite drag coefficient.

The capability of the designed system to perform a quick and accurate detection, isolation and estimation of faults affecting both the attitude control actuators, the flywheel spin rate sensors and attitude and angular velocity sensors of the spacecraft have been demonstrated. This results to be a key point in order to guarantee the desired attitude control performances through the subsequent fault accommodation in the proposed AFTC schemes. Several simulation results have been given for different fault scenarios for both the actuators and sensors. The obtained results have highlighted that the proposed diagnosis scheme can deal with the most significant, generic types of faults.

10 PUBLICATIONS

Baldi, P., Blanke, M., Castaldi, P., Mimmo, N. and Simani, S. (2015). Combined geometric and neural network approach to generic fault diagnosis in satellite reaction wheels. Accepted to the *9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes - SAFEPROCESS 2015*, Paris, France, 2-4 September 2015.

Baldi, P., Castaldi, P. and Simani, S. (2010a). Fault diagnosis and control reconfiguration in Earth satellite model engines. *Proceedings of the 9th UKACC International Conference on Control 2010*, Coventry, UK, 7-10 September 2010, pp. 114-119.

Baldi, P., Castaldi, P., Mimmo, N. and Simani, S. (2012). Aerodynamic decoupled FDI for frequency faults in earth satellite engines. *Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes - SAFEPROCESS 2012*, Mexico City, Mexico, 29-31 August 2012, vol. 8, no. 1, pp. 1095-1100.

Baldi, P., Castaldi, P., Mimmo, N. and Simani, S. (2013). Satellite attitude active FTC based on geometric approach and RBF neural network. *Proceedings of the 2nd International Conference on Control and Fault–Tolerant Systems - SysTol'13*, Nice, France, 9-11 October 2013, pp. 667-637.

Baldi, P., Castaldi, P., Mimmo, N. and Simani, S. (2014a). Generic wind estimation and compensation based on NLGA and RBF–NN. *13th European Control Conference - ECC14*, Strasbourg, France, 24-27 June 2014, pp. 1729-1734.

Baldi, P., Castaldi, P., Mimmo, N. and Simani, S. (2014b). A new aerodynamic decoupled frequential FDIR methodology for satellite actuator faults. *International Journal of Adaptive Control and Signal Processing*, September 2014, vol. 28, no. 9, pp. 812-832.

Baldi, P., Castaldi, P., Mimmo, N., A. Torre and Simani, S. (2011). A new longitudinal flight path control with adaptive wind shear estimation and compensation. *Proceedings of the 2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC),* Orlando, FL, USA, 12-15 December 2011, vol. 6, pp. 6852-6857.

Baldi, P., Castaldi, P., Simani, S. and Bertoni, G. (2010b). Fault diagnosis and control reconfiguration for satellite reaction wheels. *Proceedings of the Conference on Control and Fault Tolerant Systems - SysTol'10,* Nice, France, 6-8 October 2010, pp. 143-148.

11 Bibliography

Bastoszewicz, A. and Zuk, J. (2010). Sliding mode control - Basic concepts and current trends. *IEEE International Symposium on Industrial Electronics (ISIE),* 4-7 July 2010, pp. 3772-3777.

Beard, R.V. (1971). *Failure accomodation in linear systems through self-reorganization*. Ph.D. dissertation, Massachusetts Institute of Technology.

Benini, M., Bonfè, M., Castaldi, P., Geri, W. and Simani, S. (2008). Design and analysis of robust fault diagnosis schemes for a simulated aircraft model. *Journal of Control Science and Engineering,* vol. 2008, 18 pages.

Bennett, S. (1998). *Model based methods for sensor fault-tolerant control of rail vehicle traction*. Ph.D. dissertation, University of Hull, Kingston upon Hull, UK.

Blanke, M., Kinnaert, M., Lunze, J. and Staroswiecki, M. (2006). *Diagnosis and Fault-tolerant Control*. Springer-Verlag Berlin Heidelberg, Germany.

Buhmann, M.D. (2003). *Radial Basis Functions: Theory and Implementations*. Cambridge University Press.

Castaldi, P., Mimmo, N., Naldi, R. and Marconi, L. (2014). Robust trajectory tracking for underactuated VTOL aerial vehicles: extended for adaptive disturbance compensation. *Proceedings of the 19th IFAC World Congress*, vol. 19, no. 1, pp. 3184-3189.

Chen, J. and Patton, R.J. (1999). *Robust Model-based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishing, Boston, Massachusetts, USA.

Chen, T. and Chen, H. (1995). Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks. *IEEE Transactions on Neural Networks,* vol. 6, no. 4, pp. 904-910.

Chow, E. and Willsky, A.S. (1984). Analytical redundancy and the design of robust failure detection systems. *IEEE Transactions on Automatic Control*, vol. 29, no. 7, pp. 603-614.

Clark, R.N. (1978). Instrument fault detection. *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, no. 3, pp. 456-465.

Clark, R.N., Fosth, D.C. and Walton, V.M. (1975). Detecting instrument malfunctions in control systems. *IEEE Transactions on Aerospace and Electronic Systems,* vol. AES-11, no. 4, pp. 465-473.

Crassidis, J.L., Vadali, S.R. and Markley, F.L. (1999). Optimal tracking of spacecraft using variable-structure control. *Proceedings of the Flight Mechanics/Estimation Theory Symposium*, NASA-Goddard Space Flight Center, Greenbelt, Maryland, USA, NASA/CP-1999-209235, pp. 201-214.

Dal, M., Teodorescu, R. (2011). Sliding mode controller gain adaptation and chattering reduction techniques for DSP-based PM DC motor drives. *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 19, no. 4, pp. 531-549.

De Persis, C. and Isidori, A. (2000). On the observability codistributions of a nonlinear system. *Systems & Control Letters*, vol. 40, no. 5, pp. 297-304.

De Persis, C. and Isidori, A. (2001). A geometric approach to nonlinear fault detection and isolation. *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 853–865.

Ding S.X. (2013). *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools (2nd ed.)*. Springer-Verlag, London, UK.

Edwards, C., Lombaerts, T. and Smaili, H. (2010). *Fault Tolerant Flight Control: A Benchmark Challenge*. Springer-Verlag Berlin Heidelberg, Germany.

Edwards, C., Spurgeon, S. K. and Patton, R.J. (2000). Sliding mode observers for fault detection and isolation. *Automatica*, vol. 36, no. 4, pp. 541-553.

Egeland, O. and Gravdahl, J.T. (2002). *Modeling and Simulation for Automatic Control*. Marine Cybernetics AS, Trondheim, Norway.

Frank, P.M. (1987). Fault diagnosis in dynamic systems via state estimation - A survey. In S. Tzafestas, M. Singh and G. Schmidt (eds), System fault diagnostics, reliability and related knowledge-based approaches, Dordrecht, D. Reidel Press, pp. 35-98.

Frank, P.M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - A survey and some new results. *Automatica*, vol. 26, no. 3, pp. 459-474.

Gao, Z.W. and Ding, S.X. (2007). Actuator fault robust estimation and fault-tolerant control for a class of nonlinear descriptor systems. *Automatica*, vol. 43, no. 5, pp. 912-920.

Gao, Z.W., Ding, S.X. and Ma, Y. (2007). Robust fault estimation approach and its application in vehicle lateral dynamic systems. *Optimal Control Applications and Methods*, vol. 28, no. 3, pp. 143-156.

Gertler, J. (1991). Analytical redundancy methods in fault detection and isolation. *Proceedings of IFAC/IAMCS Symposium on SAFEPROCESS'91*, Baden-Baden, vol. 1, pp. 9-21.

Gertler, J. (1998). *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, New York.

Hughes, P.C. (1986). *Spacecraft Attitude Dynamics*. John Wiley & Sons Inc., New York.

Hwang, I., Kim, S., Kim, Y. and Seah, C.E. (2010). A survey of fault detection, isolation, and reconfiguration methods. *IEEE Transactions on Control Systems Technology*, vol. 18, no. 3, pp. 636-653.

Ibrahim, A.El-S., Tobal, A.M. and Sultan, M.A. (2012). Satellite attitude maneuver using sliding mode control under body angular velocity constraints. *International Journal of Computer Applications*, vol. 50, no. 13, pp. 41-46.

Isermann, R. (1984). Process fault detection based on modeling and estimation methods - A survey. *Automatica*, vol. 20, no. 4, pp. 387-404.

Isermann, R. (2005). Model-based fault detection and diagnosis - Status and applications. *Annual Reviews in Control*, vol. 29, no. 1, pp. 71-85.

Isermann, R. (2006). *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer-Verlag Berlin Heidelberg, Germany.

Isermann, R. (2011). *Fault-Diagnosis Applications - Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault-Tolerant Systems*. Springer-Verlag Berlin Heidelberg, Germany.

Isermann, R. and Ballé, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, vol. 5, no. 5, pp. 709-719.

Isidori, A. (1995). *Nonlinear Control Systems (3rd ed.)*. Springer-Verlag, London, UK.

Isidori, A., Krener, A.J., Gori-Giorgi, C., Monaco, S. (1981). Nonlinear decoupling via feedback: a differential geometric approach. *IEEE Transactions on Automatic Control*, vol. AC-26, no. 2, pp. 331-345.

Jiang, B., Staroswiecki, M. and Cocquempot, V. (2004). Fault estimation in nonlinear uncertain systems using robust/sliding-mode observers. *IET Proceedings on Control Theory & Applications*, vol. 151, no. 1, pp. 29-37.

Kaplan, M.H (1976). *Modern Spacecraft Dynamics and Control*. John Wiley & Sons Inc., New York.

Lunze, J. and Richter, J.H. (2008). Reconfigurable fault-tolerant control: a tutorial introduction. *European Journal of Control*, vol. 14, no. 5, pp. 359-386.

Massoumnia, M.A. (1986). A geometric approach to the synthesis of failure detection filters. *IEEE Transactions on Automatic Control*, vol. AC-31, no. 9, pp. 839-846.

Massoumnia, M.A., Verghese, G.C. and Willsky, A.S. (1989). Failure detection and identification. *IEEE Transactions on Automatic Control*, vol. AC-34, no. 3, 316-321.

Mattone, R. and De Luca, A. (2006a). Relaxed fault detection and isolation: an application to a nonlinear case study. *Automatica*, vol. 42, no. 1, pp. 109–116.

Mattone, R. and De Luca, A. (2006b). Nonlinear fault detection and isolation in a three-tank system. *IEEE Transactions on Control System Technology*, vol. 14, no. 6, pp. 1158-1166.

Morse, A.S. and Wonham, W.M. (1970). Decoupling and pole assignment by dynamic compensation. *SIAM Journal on Control*, vol. 8, no. 1, pp. 317-337.

Morse, A.S. and Wonham, W.M. (1971). Status of noninteracting control. *IEEE Transactions on Automatic Control*, vol. AC-16, no. 6, pp. 568-581.

Nazari, R., Seron, M.M. and De Doná, J.A. (2013). Fault-tolerant control of systems with convex polytopic linear parameter varying model uncertainty using virtual-sensor-based controller reconfiguration. *Annual Reviews in Control*, vol. 37, no. 1, pp. 146-153.

Noura, H., Sauter, D., Hamelin, F. and Theilliol, D. (2000). Fault-tolerant control in dynamic systems: application to a winding machine. *IEEE Control Systems*, vol. 20, no. 1, pp. 33-49.

Noura, H., Theilliol, D., Ponsart, J. and Chamseddine, A. (2009). *Fault-tolerant Control Systems: Design and Practical Applications*. Springer-Verlag, London, UK.

Park, J., Sandberg, I.W. (1991). Universal approximation using radial-basis-function networks. *Neural Computation*, vol. 3, no. 2, pp. 246 - 257.

Patton, R.J. (1997a). Fault-tolerant control systems: the 1997 situation. *Proceedings of the 3rd IFAC Symposium on SAFEPROCESS'97*, Hull, vol. 2, pp. 1033-1054.

Patton, R.J. (1997b). Robustness in model-based fault diagnosis: the 1995 situation. *Annual Reviews in Control*, vol. 21, pp. 103-123.

Patton, R.J. and Chen, J. (1991). A re-examination of the relationship between parity space and observer-based approaches in fault diagnosis. *European Journal of Diagnosis and Safety in Automation*, vol. 1, no. 2, pp. 183-200.

Patton, R.J. and Chen, J. (1994). A review of parity space approaches to fault diagnosis for aerospace systems. *Journal of Guidance, Control & Dynamics*, vol. 17, no. 2, pp. 278-285.

Patton, R.J., Frank, P.M. and Clark, R.N. (1989). *Fault Diagnosis in Dynamic Systems: Theory and Application*. Control Engineering Series, Prentice-Hall, New York.

Patton, R.J., Frank, P.M. and Clark, R.N. (2000). *Issues of fault diagnosis for dynamic systems*. Springer-Verlag, London, UK.

Ponsart, J.C., Theilliol, D. and Aubrun, C. (2010). Virtual sensors design for active fault tolerant control system applied to a winding machine. *Control Engineering Practice*, vol. 18, no. 9, pp. 1037-1044.

Richter, J.H. (2011). *Reconfigurable Control of Nonlinear Dynamical Systems: A Fault-hiding Approach*. Springer-Verlag Berlin Heidelberg, Germany.

Rothenhagen, K. and Fuchs, F.W. (2009). Doubly fed induction generator model-based sensor fault detection and control loop reconfiguration. *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4229-4238.

Shi, F. (2013). *Observer based active fault tolerant control of descriptor systems*. Ph.D. dissertation, University of Hull.

Sidi, M.J. (1997). *Spacecraft Dynamics and Control: A Practical Engineering Approach*. Cambridge University Press.

Simani, S., Fantuzzi, C. and Patton, R.J. (2003). *Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques*. Springer-Verlag, London, UK.

Slotine, J.J.E. and Li, W. (1991). *Applied Nonlinear Control*. Prentice-Hall, Englewood Cliffs, NJ.

Sontag, E.D. and Wang, Y. (1996). New characterizations of input-to-state stability. *IEEE Transactions on Automatic Control*, vol. 41, no. 9, pp. 1283-1294.

Sun, X. (2013). *Unknown input observer approaches to robust fault diagnosis*. Ph.D. dissertation, University of Hull.

Tewari, A. (2007). *Atmospheric and Space Flight Dynamics: Modeling and Simulation with MATLAB and Simulink*. Birkhäuser, Boston, Massachusetts, USA.

Wang, Z., Shen, Y. and Zhang, X. (2011). Actuator fault detection and estimation for a class of nonlinear systems. *7th International Conference on Natural Computation (ICNC)*, Shanghai, China, 26-28 July 2011, vol. 1, pp. 535-539.

Wertz, J.R. (1978). *Spacecraft Attitude Determination and Control*. D. Reidel Publishing Company, Dordrech, Netherlands.

Wertz, J.R. and W.J. Larson (1999). *Space Mission Analysis and Design (3rd ed.)*. Microcosm Press and Kluwer Academic Publishers.

Wie, B., (2008). *Space vehicle dynamics and control (2nd ed.)*. American Institute of Aeronautics and Astronautics.

Willems, J.C. and Commault, C. (1981). Disturbance decoupling by measurement feedback with stability or pole placement. *SIAM Journal on Control and Optimization*, vol. 19, no. 4, pp. 490-504.

Willsky, A. (1976). A survey of design methods for failure detection systems. *Automatica*, vol. 12, pp. 601-611.

Witczak, M. (2014). *Fault Diagnosis and Fault-Tolerant Control Strategies for Non-Linear Systems: Analytical and Soft Computing Approaches*. Springer International Publishing.

Wu, N.E., Thavamani, S., Zhang, Y. and Blanke, M. (2006). Sensor fault masking of a ship propulsion system. *Control Engineering Practice*, vol. 14, no. 11, pp. 1337-1345.

Wunnenberg, J. (1990). *Observer-based fault detection in dynamic systems*. Ph.D. dissertation, University of Duisburg.

Young, K.D. and Utkin, V.I. (1999). A control engineer’s guide to sliding mode control. *IEEE Transactions on Control Systems Technology*, vol. 7, no. 3, pp. 328-342.

Zhang, H. (2009). *Software sensors and their applications in bioprocess. Computational Intelligence Techniques for Bioprocess Modelling, Supervision and Control*. Springer-Verlag Berlin Heidelberg, Germany.

Zhang, K., Jiang, B. and Shi, P. (2009). Fast fault estimation and accommodation for dynamical systems. IET Control Theory & Applications, vol. 3, no. 2, pp. 189-199.

Zhang, Y. and Jiang, J. (2008). Bibliographical review on reconfigurable fault–tolerant control systems. *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252.