A Multiple-Observer Scheme for Fault Detection, Isolation and Recovery of Satellite Thrusters

Antoine Abauzit, Julien Marzat

A Multiple-Observer Scheme for Fault Detection, Isolation and Recovery of Satellite Thrusters

Antoine Abauzit and Julien Marzat

Abstract The method proposed in this paper aims at automatically detecting, isolating and identifying faults on actuators of a satellite model and also aims at automatically reconfiguring the reference input once the fault has been isolated. The method uses two sliding mode observers to detect and reconstruct the fault. A cusum test on the output of the detection observer triggers a bank of Unknown Input Observers in order to isolate the faulty actuator. The reference input is automatically reconfigured in order to pre-compensate the fault, which makes the satellite capable of fulfilling its mission with the desired performances and good precision. Monte Carlo analysis, based on performance criteria, is carried out to assess the performance of the strategy. The combination of these different types of filters might provide better detection, isolation and identification capabilities than a single filter that would be forced to achieve a trade-off between fast detection and accurate estimation.

1 Introduction

During the last decades, fault detection, isolation and recovery (FDIR) has met a growing interest in the scientific community. The higher levels of automation expected from modern systems require a higher reliability. Hardware redundancy is usually employed to achieve this reliability yet it implies added complexity and higher costs. For satellite systems, hardware redundancy is particularly cumbersome since each actuator should be built several times, and the cumulated mass leads to a much higher launch cost. Furthermore, it is almost impossible to repair

Antoine Abauzit
ONERA – The French Aerospace Lab, F-91123 Palaiseau, France
e-mail: antoine.abauzit@onera.fr

Julien Marzat
ONERA – The French Aerospace Lab, F-91123 Palaiseau, France
e-mail: julien.marzat@onera.fr

the satellite if a fault occurs after launch. Therefore, methods avoiding hardware redundancy and making it possible for the satellite to fulfil its mission in spite of a fault are highly recommended.

By using a mathematical model of the system, model-based fault diagnosis (also called *analytical redundancy*) can be able to detect and isolate faults on actuators or sensors. Thus, the reliability of the system can rely less heavily on hardware redundancy and more on software efficiency.

Different model-based methods have been studied for satellite models: parity space [10], neural-network [8], parameter estimation techniques [4], observer-based techniques [1], bank of Kalman filters [12] and techniques based on unknown input observer (UIO) [6].

When a fault occurs, the system has to be able to carry on its mission. A Fault-Tolerant Control System (FTCS) is a closed-loop system that has the ability to tolerate faults without threatening its performances or its stability. The reader may refer to the excellent bibliographical review [13], which explains the existing approaches on this topic.

The method presented in this paper aims at automatically detecting a fault on an actuator of a satellite model, isolate it, and reconfigure the control input in order to carry on the mission. This allows the assessment of the performance of a whole FDIR loop on a realistic aerospace model, which is seldom addressed in the literature. The detection and the reconstruction of the fault are achieved via sliding mode observers like in [3]. The obtained residuals are analyzed by cusum tests in a decision-making scheme. Once the fault is detected, a bank of UIOs isolates the actuator on which the fault occurred. An interesting feature of the proposed method is to make use of observers with different dynamics for the detection and estimation of faults to escape from the classical trade-off between reliable detection (few false alarms) and fast estimation. Finally, the reference input is modified in order to compensate for the effect of the fault like in [9]. This way, the dynamics of the feedback laws remain unchanged and a good precision can be achieved, without the need to reconfigure entirely the controller. The performance of the fault diagnosis method is evaluated with Monte-Carlo (MC) simulations.

2 Satellite Modelling

2.1 Satellite Dynamics

The vehicle considered is a deep-space satellite with 12 thrusters (organized in 4 sets), similar to the one presented in [12]. The state of the satellite is described in Eq. (2.1). $\mathbf{p}$ is the inertial position, $\mathbf{v}$, the inertial velocity, $\mathbf{q}$ the quaternion describing the rotation from the inertial frame to the body frame, and $\boldsymbol{\omega}$ the angular velocity of the satellite.

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \boldsymbol{\omega} \end{bmatrix} \tag{2.1}$$

The sensitivity matrix $\mathbf{B_F}$ represents the force due to each thruster input. The j^{th} column of $\mathbf{B_F}$ refers to the direction of the j^{th} thrust of the satellite, $\mathbf{d}_j$, described in the body frame.

$$\mathbf{B_F} = [\mathbf{d}_1 \;\; \mathbf{d}_2 \;\; \cdots \;\; \mathbf{d}_{12}] \tag{2.2}$$

$$\mathbf{B_F} = \begin{bmatrix} 1 & 1 & 1 & 1 & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0 & 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{3}}{2} & 0 & 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0 & 0 & -\frac{\sqrt{3}}{2} \end{bmatrix}$$

The input from the thrusters is defined by the vector **u**. Thereby the input of the i^{th} thruster is the i^{th} component of the vector, u_i. It is a positive scalar value between 0 and 100 N.

In the end, the net force of all the thrusters, in the body frame, simply is:

$$\mathbf{F} = \mathbf{B_F}\mathbf{u} = \sum_{i=1}^{12} \mathbf{d}_i u_i \tag{2.3}$$

The dynamics of the inertial position and the inertial velocity are described in Eq. (2.4):

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \frac{1}{m}\mathbf{R}_{\text{B}\to\text{I}}\mathbf{B_F}\mathbf{u} \end{aligned} \tag{2.4}$$

where $\mathbf{R}_{\text{B}\to\text{I}}$ is the rotation matrix that turns a vector described in the body frame into a vector described in the inertial frame. The mass of the vehicle is denoted by m.

The rotational dynamics are generally described by [7]:

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix}\mathbf{q} \tag{2.5}$$

$$\boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.6}$$

$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1}\mathbf{B}_{\mathbf{T}}\mathbf{u} - \mathbf{I}^{-1}\boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \tag{2.7}$$

The inertia of the satellite in the body frame is defined as **I**. The sensitivity matrix $\mathbf{B_T}$ represents the torque of each thruster input. The j^{th} column of $\mathbf{B_T}$ refers to the direction of the torque due to the j^{th} thruster in the body frame.

$$\mathbf{B}_{\mathbf{T}} = [\mathbf{d}_{\mathrm{T1}} \;\; \mathbf{d}_{\mathrm{T2}} \;\; \cdots \;\; \mathbf{d}_{\mathrm{T12}}] \tag{2.8}$$

with $\mathbf{d}_{\mathrm{Ti}} = \mathbf{G}\mathbf{A}_{\mathrm{i}} \times \mathbf{d}_{\mathrm{i}}$ where A_i is the point where the thrust applies and G is the center of mass of the satellite.

Table 2.1 Description of the parameters of the satellite model (from [12])

Parameter	Value	Unit
Mass	879	[kg]
Ixx	2787	[kg.m²]
Iyy	2836	[kg.m²]
Izz	2266	[kg.m²]

2.2 Control Allocation

Two state feedback laws are used to control the state of the satellite. The first one is dedicated to the control of the position and the velocity while the second one is for the attitude and the angular velocity. The outputs of the linear controller and the attitude controller are respectively linear and angular accelerations.

A nonlinear iterative control allocation procedure [11] is used to compute the reference thrust for each actuator, in order to respect the commands coming from the feedback laws. The output of the allocator is a desired thrust level for each actuator, which is assumed to be achieved instantaneously.

2.3 Measurement Model

Star trackers assumed to be faultless measure the attitude of the satellite relative to the inertial frame. Let **q** be the actual attitude of the satellite. The measurement of the star trackers is corrupted by a rotation error $\mathbf{q_{err}}$. The measured quaternion $\mathbf{q_m}$ is given by the quaternion product:

$$\mathbf{q}_{\mathrm{m}} = \mathbf{q}_{\mathrm{err}} \otimes \mathbf{q} \tag{2.9}$$

For the measurement of the angular velocity, the quaternion between the previous measured quaternion $\mathbf{q_p}$ and the current measured quaternion is first computed:

$$\boldsymbol{\delta q} = \mathbf{q}_{\mathrm{p}}^{-1} \otimes \mathbf{q}_{\mathrm{m}} \tag{2.10}$$

The attitude variation is:

$$\boldsymbol{\delta q} = \begin{bmatrix} \cos\dfrac{\varphi}{2} \\ \sin\dfrac{\varphi}{2}\mathbf{e} \end{bmatrix} \tag{2.11}$$

where **e** is a unit vector that gives the axis of the rotation. In the end, we have an angular increment of φ around the axis **e** between two steps.

The angular velocity estimation is then:

$$\boldsymbol{\omega}_{\mathbf{est}} = \frac{\varphi}{\Delta t}\mathbf{e} \tag{2.12}$$

where Δt is the time step of the control system. In the following, we assume that the measurement of the angular velocity is given by $\boldsymbol{\omega}_{\mathbf{mes}} = \boldsymbol{\omega}_{\mathbf{est}}$. It is of course worth mentioning that more refined methods may lead to a better estimation of the angular velocity, yet it seemed to be a sufficient modelling level to assess the interest of the proposed method.

3 FDIR Methodology

The objective of the proposed method is to quickly detect, then isolate, and finally compensate a fault on a thruster when it happens. Since two thrusters can have the same force direction but different torque directions, we assume that it is easier to get explicit residuals from the angular velocity than from the linear velocity. If two thrusters have the same force and torque directions, it is necessary to study the linear velocity too in order to determine the sign of the fault and isolate the faulty actuator.

Two sliding mode observers are used to detect and to reconstruct the fault signals. The design of these observers is based on [3]. This kind of non-linear observer generates an output estimate $\hat{\mathbf{y}}$ and a state estimate $\hat{\mathbf{x}}$ such that the estimation error converges to zero in finite time. In [3], this observer is written in the form:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} - \mathbf{G}_{\mathbf{l}}\mathbf{e}_{\mathbf{y}} + \mathbf{G}_{\mathbf{n}}\mathbf{v} \tag{3.1}$$

where $\mathbf{e}_y = \hat{\mathbf{y}} - \mathbf{y}$ is the output error, $\mathbf{v}$ is a non-linear switched function of the output error and $\mathbf{G}_l$ and $\mathbf{G}_n$ are gain matrices. It is shown that once the sliding motion $\mathbf{e}_y = \mathbf{0}$ and $\dot{\mathbf{e}}_y = \mathbf{0}$ is attained, it becomes possible to estimate actuator or sensor faults from the *equivalent output injection* signal required to maintain sliding motion.

The two observers only differ in their tuning since opposite properties are required for them: one has to provide explicit residuals while the other has to quickly reconstruct the disturbance.

Once a detection flag has been raised, a bank of Unknown Input Observers (UIO) is used to isolate the faulty thruster as in [6].

All these observers only reconstruct the angular velocity of the satellite from $\boldsymbol{\omega}_{\mathbf{mes}}$ and the satellite model. The computational cost is then reduced, since there is no need to process the entire state vector.

Figure 1 depicts the interaction between the satellite dynamics and the on-board algorithms including navigation, state feedback, the control allocation procedure, the FDI functions and the reconfiguration of the input command.

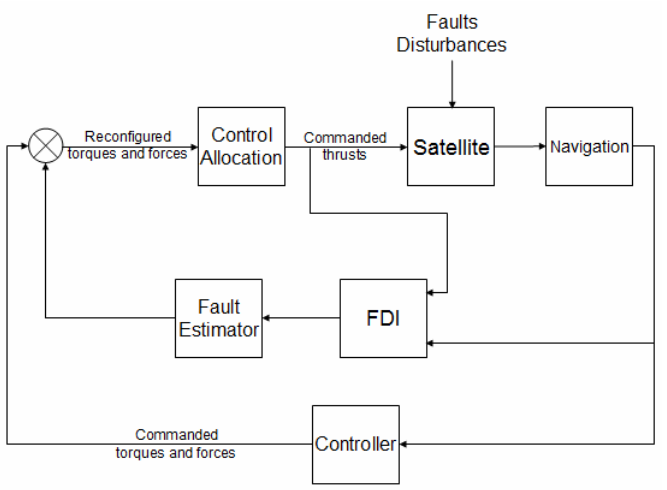

Fig. 1. Control system and FDIR methodology

Figure 2 is a detailed illustration of the FDI block from the previous diagram. It depicts the faulty actuator isolation process and the estimation of the torque disturbance due to the fault.

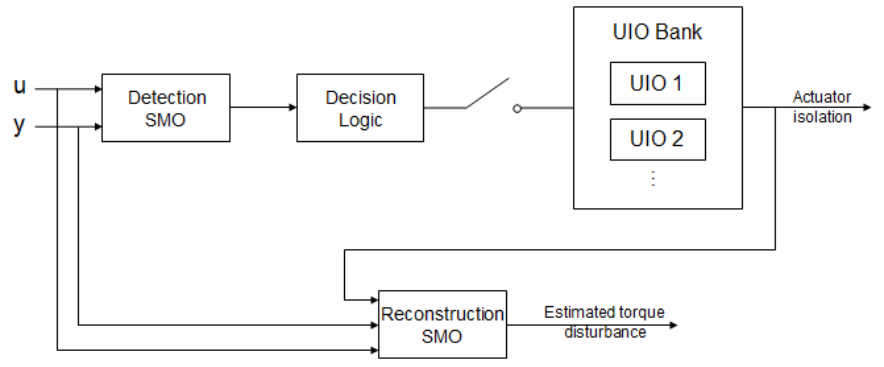

Fig. 2. Detailed FDI scheme

3.1 Detection Observer

In order to decide if a fault happened or not, a detection observer is designed. For computational complexity, the detection observer is the same as the reconstruction observer described further, but with a different tuning.

The dynamics of this observer are chosen to be slow, as a result, when a fault occurs, the error between the measured angular velocity and the output of the observer quickly increases.

3.2 A Cusum Test to Trigger the Isolation Process

The decision is based on a cusum test [2] on the angular velocity estimation error. The mean error of this state is supposed to be small when no fault occurs. If a fault happens, the mean value of the estimation error evolves. When a threshold is crossed on the cusum statistics, the test indicates that an actuator may be faulty. It is then time to isolate the fault.

3.3 Fault Isolation: a Bank of UIOs

For real-time applicability, the UIOs are triggered only when the decision criterion indicates that a fault has occurred. Even if only the angular velocity is estimated, keeping the UIOs switched off before the fault is detected seems to be a good strategy, regarding the number of actuators of the satellite studied here.

The isolation of the actuator is based on a bank of observers like in [6]. Unknown Input observers have been chosen here because of their decoupling capabilities. Once again, the angular velocity measurement is used by the observers, but this time for isolation.

For each thruster, an UIO is designed. The tuning is the same for all the actuators. Each UIO is such that it can fully reconstruct the angular velocity with all the

inputs but one. As a result, the UIO dedicated to the faulty actuator will not be affected by the fault while all the others will be.

A UIO can be written in the form:

$$\begin{aligned} \dot{\mathbf{z}} &= \mathbf{Fz} + \mathbf{TBu} + \mathbf{Ly} \\ \hat{\mathbf{x}} &= \mathbf{z} + \mathbf{Hy} \end{aligned} \tag{3.2}$$

For the state to be observable despite an unknown input, the design of the UIO has to respect some constraints.

In our case, the output equation of the state space representation simply is:

$$\mathbf{y} = \mathbf{Cx} = \mathbf{x}, \text{ i.e. } \mathbf{C} = \mathbf{I_3} \tag{3.3}$$

Thus, for each UIO, the matrix **H** is such than $\mathbf{T} = \mathbf{I_3}\text{-}\mathbf{H}$ is orthogonal to the direction of the "missing" thruster, i.e.:

$$(\mathbf{I_3}\text{-}\mathbf{H})\mathbf{d}_{Ti} = \mathbf{0} \tag{3.4}$$

Here, the matrix **F** is very simple:

$$\mathbf{F} = -\mathbf{K} \tag{3.5}$$

where **K** is a feedback matrix designed by pole placement. This matrix should be stable in order to ensure the convergence of the estimation error $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$.

Finally, the last condition to ensure the convergence is to design the matrix **L** as follows:

$$\mathbf{L} = \mathbf{K}(\mathbf{I_3}\text{-}\mathbf{H}) \tag{3.6}$$

Since we are not trying to reconstruct the fault here – we just want to isolate the faulty actuator – a classical observer design is proposed for the UIOs.

The UIO bank has to indicate clearly which actuator is faulty. Since the UIO dedicated to the faulty actuator is insensitive to the fault, the residuals of the other UIOs have to be very sensitive to input errors. The tuning of the UIO is such that the error estimation rapidly increases when the fault appears, while the residuals on the faulty UIO stay small. One drawback of this method is that even if an UIO can fully reconstruct the state, it can take time to have a small estimation error in the direction of the "missing" actuator.

To overcome this drawback, the residuals on the angular velocity are described in a specific frame for each UIO: the rotation matrix is such that the first axis of the new frame is the direction of the torque of the dedicated actuator. This means that the transient phase only happens on the first component while the others are quickly fully reconstructed. This is closely related to projections used in the context of parity space techniques for enhancing sensitivity to faults.

As a result, the residuals on the first component of each UIO will always converge to 0, with or without fault. On the other hand, the residuals on components 2 and 3 of all the UIO but the faulty one will increase when the fault appears.

It is now possible to study the norm of the residuals on components 2 and 3. The UIO that minimizes the norm of the residual on components 2 and 3 indicates which actuator is faulty.

3.4 Disambiguation Process

The UIOs use the direction of the torque due to the thrusters to isolate the faulty actuator. However, among the 12 thrusters of the satellite, two pairs of actuators have opposite directions making a disambiguation process necessary if the fault happens on one of these thrusters. We have:

$$\begin{aligned} \mathbf{d}_{T3} &= -\mathbf{d}_{T1} \\ \mathbf{d}_{T4} &= -\mathbf{d}_{T2} \end{aligned} \tag{3.7}$$

As a result, isolating the faulty thruster between actuators 1 and 3 or 2 and 4 requires the knowledge of the sign of the fault.

These four thrusters have the same force direction that is, described in the body frame:

$$\mathbf{d}_1 = \mathbf{d}_2 = \mathbf{d}_3 = \mathbf{d}_4 \tag{3.8}$$

The sign of the fault can be determined by comparing the commanded linear acceleration (i.e. the output of the control law on position and velocity) with the estimate of the linear acceleration computed from velocity measurement. Once the sign is known, the fault is easily isolated.

3.5 Reconstruction Observer

The design of this observer is based on the sliding mode observer proposed by [3] where the sliding motion is maintained even in the presence of faults. An estimate of the fault can be computed thanks to the *equivalent output injection* signal that maintains the sliding motion.

The reconstruction observer proposed here only deals with the angular velocity of the satellite. The state estimation follows the dynamics:

$$\dot{\hat{\boldsymbol{\omega}}} = \mathbf{I}^{-1}\mathbf{B}_\mathbf{T}\mathbf{u} - \mathbf{I}^{-1}\mathbf{y}\times\mathbf{I}\mathbf{y} + \mathbf{W}\mathbf{e}_\mathbf{y} + \mathbf{v} \tag{3.9}$$

where $\mathbf{W}$ is a stable design matrix and the vector $\mathbf{v}$ is defined as:

$$\mathbf{v} = -M \arctan \mathbf{e}_\mathbf{y} \text{ with } M>0 \tag{3.10}$$

When an additive fault $\boldsymbol{\Delta}\mathbf{u}$ occurs, the dynamics of the angular velocity become:

$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1}\mathbf{B}_\mathbf{T}(\mathbf{u} + \boldsymbol{\Delta}\mathbf{u}) - \mathbf{I}^{-1}\boldsymbol{\omega}\times\mathbf{I}\boldsymbol{\omega} \tag{3.11}$$

The state estimation error is defined as $\mathbf{e}_y = \hat{\boldsymbol{\omega}} - \boldsymbol{\omega}$, so its dynamics are:

$$\dot{\mathbf{e}}_y = \mathbf{W}\mathbf{e}_y + \mathbf{v} - \mathbf{I}^{-1}\mathbf{B}_T\Delta\mathbf{u} - \mathbf{I}^{-1}\mathbf{y}\times\mathbf{I}\mathbf{y} + \mathbf{I}^{-1}\boldsymbol{\omega}\times\mathbf{I}\boldsymbol{\omega} \tag{3.12}$$

It has been shown by [3] that during the sliding motion, $\mathbf{e}_y = \mathbf{0}$ and $\dot{\mathbf{e}}_y = \mathbf{0}$, so if no fault happens, $\mathbf{v}\rightarrow 0$ and if a fault occurs $\mathbf{v}\rightarrow \mathbf{I}^{-1}\mathbf{B}_T\Delta\mathbf{u}$.

Thanks to the reconstruction observer, it is possible to compute an estimation of the torque disturbance due to the fault. If the fault has been correctly isolated, it is now possible to estimate it.

3.5 Reconfiguration of the Input Command

Once that the faulty actuator has been isolated and that the torque disturbance due to the fault has been reconstructed, it is possible to compute the value of the fault on the actuator thanks to the method of least squares.

Let $\mathbf{F}_{\mathbf{pert}}$ and $\mathbf{T}_{\mathbf{pert}}$ be the estimation of the disturbance due to the actuator fault on the force and the torque respectively, and let $\mathbf{F}_{\mathbf{co}}$ and $\mathbf{T}_{\mathbf{co}}$ the output of the control laws.

Once that a fault is detected, the input of the allocator becomes:

$$\begin{aligned} \mathbf{F}_{\text{alloc}} &= \mathbf{F}_{\text{co}} - \mathbf{F}_{\text{pert}} \\ \mathbf{T}_{\text{alloc}} &= \mathbf{T}_{\text{co}} - \mathbf{T}_{\text{pert}} \end{aligned} \tag{3.13}$$

The disturbance is directly pre-compensated in the control laws; therefore, even if the disturbance is not perfectly rejected because of the model errors, the "actual" commands $\mathbf{F}_{\mathbf{co}}$ and $\mathbf{T}_{\mathbf{co}}$ can be respected.

4 Simulations

4.1 An Example

The proposed method has been applied to the satellite model described in Section 2.1. The method is first illustrated on one example. For our scenario, the faulty actuator is the number 12. Figure 3 shows the residuals of the detection observer. The residuals clearly increase after the occurrence of the fault. The cusum test on the residuals of the detection observer triggers the bank of UIO.

The UIO dedicated to the 12^{th} actuator is insensitive to faults on this actuator. Its residuals remain close to zero, despite the model errors that we considered here, while the residuals of the 11^{th} UIO quickly increase (Figure 4). The UIO that minimizes the residuals on components 2 and 3 indicates which actuator is the faulty one. The faulty actuator is automatically detected and isolated (Figure 5).

The amplitude of the fault is estimated (Figure 6) then pre-compensated by the re-configuration of the reference input.

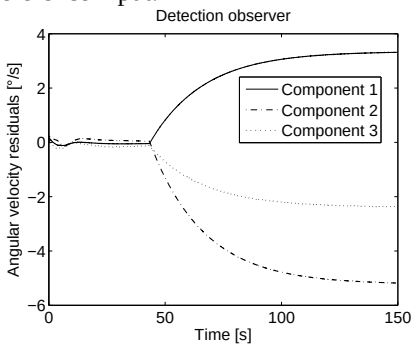

Fig. 3. Output of the detection observer

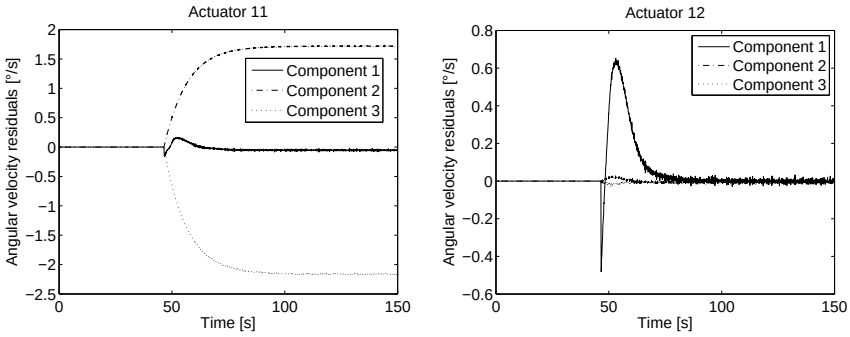

Fig. 4. Residuals of the 11th and the 12th UIOs

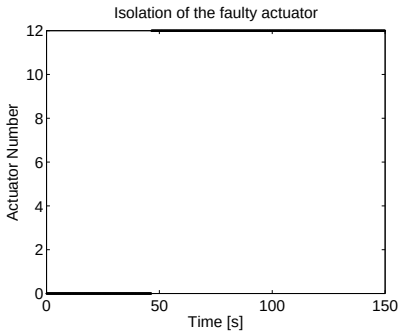

Fig. 5. Index of the faulty actuator

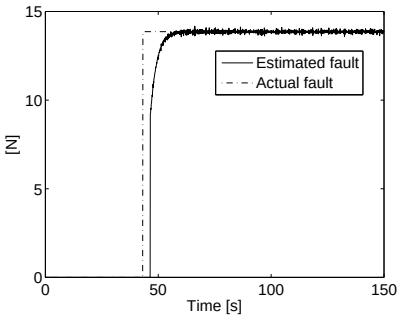

Fig. 6. Fault estimation

4.2 Influence of the Tuning of the SMOs

In order to highlight the interest of using different tunings for the detection and the reconstruction observers, the previous example is run again twice. The difference of tuning between the detection and the reconstruction observers lies in the scalar parameter M from equation 3.10. Let M_D be the parameter for the detection observer and M_R the parameter for the reconstruction observer.

In the tuning presented above, the mixed tuning, the parameter M_R is equal to ten times the parameter M_D since the dynamics of the detection are chosen to be slow while fast dynamics are required to estimate the fault quickly. The tuning where the detection and the reconstruction observers both have slow dynamics is referred as case 1 in the table below while the tuning where both have fast dynamics is referred as case 2.

Table 4.1 SMOs parameter M tuning

	M_D	M_R
Case 1: slow tuning	M_1	M_1
Case 2: fast tuning	10 M_1	10 M_1
Mixed tuning	M_1	10 M_1

Since the detection observer is the same for the case 1 and the mixed tuning, the detection happens at the same date. However, in case 1, the fault estimation converges much slower to the actual fault (Figure 7) because of the slow dynamics of the reconstruction observer. As a result, the input command can not be reconfigured efficiently.

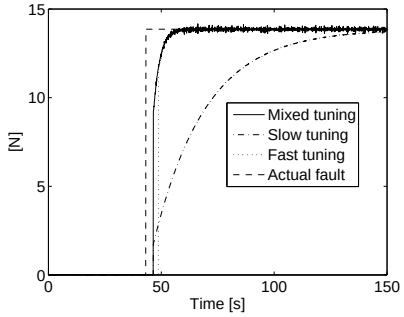

Fig. 7. Fault estimation for the three tunings

For the case 2, the fault estimation is as fast as with the mixed tuning since the reconstruction observers are the same. Unfortunately, the detection time is longer than with the mixed tuning, by 2.3 s on this example. Indeed, once the fault has appeared, the residuals of the detection observer do not grow as fast as with the mixed tuning. As a result, it takes more time to cross the threshold of the cusum test making the isolation process triggered later. Furthermore, since the final val-

ues of the residuals are smaller, it becomes harder to detect small faults (Figure 8 compared to Figure 3) and the missed fault rate would be higher.

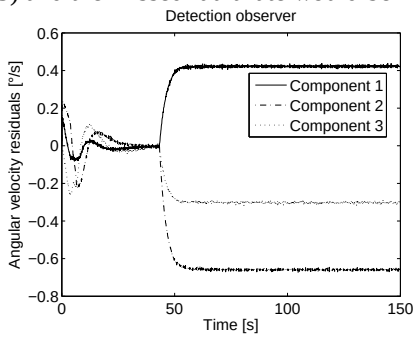

Fig. 8. Output of the detection observer for the fast tuning (case 2)

The tuning of the cusum test could be adapted to the fast tuning of the detection observer, but there is a risk that the residuals disturbed by the model errors would trigger the detection observer, especially in the first seconds of flight, resulting in an increased false alarm rate.

In the end, the use of two SMOs, one with slow dynamics dedicated to the detection, and one with fast dynamics dedicated to the fault estimation makes it possible to circumvent the usually required trade-off between the two tunings.

4.3 Performance Evaluation

In order to evaluate the efficiency of the method and its robustness to model errors, some performance indices should be defined. The fault isolation time indicates the performance of the method while the missed fault rate and the false alarm rate allow the evaluation of its robustness.

The fault isolation time is the difference between the moment the isolation is definitively done (the actuator indicator remains constant until the end of the scenario) and the moment the fault occurs.

The false alarm rate is the number of times a fault is detected while all the actuators are still healthy divided by the total number of scenarios.

The missed fault rate is the number of time a fault was not detected at the end of the scenario divided by the total number of scenarios.

To compute these indices, Monte-Carlo simulation tests have been carried out. For each simulation run, parameters of the model such as the position of the center of mass, the mass, the inertia, and the direction of the thrusters are altered. The value of the fault, the time of its appearance, and the index of the faulty actuator are randomly chosen.

All the model alterations follow a uniform distribution with bounds as described in Table 4.2.

Table 4.2 Summary table of the altered parameters of the satellite and their distribution

Parameter	Unit	Lower bound – Upper bound
Mass	[kg]	-10; +10
Inertia (by axis)	[kg.m²]	-10; +10
Center of mass (by axis)	[m]	-0.02; +0.02
Thruster direction (by actuator)	[°]	-0.2; +0.2

All the fault parameters follow a uniform distribution with bounds as described in Table 4.3. For each scenario, after the occurrence time, a bias appears on the faulty actuator until the end of the simulation.

Table 4.3 Summary table of the fault parameters

Parameter	Unit	Lower bound – Upper bound
Value	[N]	-30; +30
Actuator	-	1; 12
Occurrence time	[s]	0; 100

A number of 1000 MC simulation tests have been carried out. The false alarm rate and the missed fault rate can be directly computed. It appears that the appropriate tuning of the method allows avoiding false alarms since the false alarm rate is equal to zero. On the other hand, 49 faults were not detected among the 1000 scenarios. This means that we have a missed fault rate of 4.9%.

It might be possible to get a lower missed fault rate with different tuning of the cusum test but it could lead to a higher false alarm rate. It should however be noted that the values of the non-detected faults are small and have thus a very small effect on the system dynamics.

Figure 9 presents the detection time for the different fault values met in the 1000 scenarios. It clearly appears that the detection time depends on the value of the fault: the more the fault is important, the faster it is isolated. The smallest faults – less than 1.5N – are not detected.

Figure 10 shows the required delay for the fault isolation once it is detected. It appears that 50 % of the faults are isolated less than 0.2s after detection, and that 88 % are isolated within the second that follows the detection. In some cases - small fault, high model errors - the isolation time can be more important, however, the isolation is always achieved in the end.

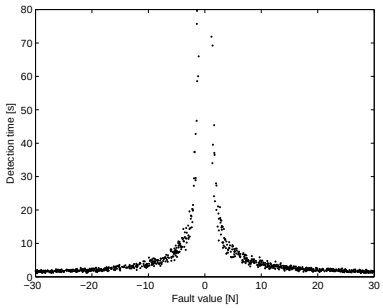

Fig. 9. Detection time versus fault

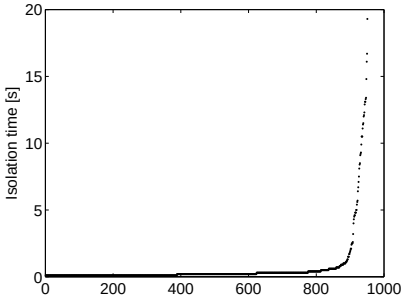

Fig. 10. Isolation time after detection

5 Conclusion

This paper has described a method to detect, isolate and compensate an actuator fault in a satellite system. The main challenge is to detect as soon as possible the fault and to reconstruct it rapidly among all the possible thrusters. The proposed method differs from the usual solutions by the use of two observers, one for detection, and one for reconstruction. The detection and the reconstruction observers use the same kind of sliding mode observer but with different tunings. The isolation of the faulty actuator is performed with a bank of UIO that is triggered by the detection observer. Each of the UIO residuals are described in a specific frame bound to the dedicated actuator in order to get more "readable" residuals for an easier isolation. Once a fault is isolated and reconstructed, the input command is reconfigured in order to pre-compensate the disturbances due to the fault.

The proposed method allows to successfully detect and isolate faults in most cases. Only the smallest faults are not detected but such faults lead to small disturbances, so the integrity of the system is not threatened. An optimal tuning thanks to a minimax optimization could allow to reduce the missed fault alarm without deteriorating the false alarm rate, taking into account the sources of uncertainty [5].

References

1. Alwi, H., Edwards, C., Marcos, A.: FDI for a Mars orbiting satellite based on a sliding mode observer scheme. In Proc. of the IEEE Conference on Control and Fault-Tolerant Systems, Nice, France, 143-148 (2010)

2. Basseville, M., Nikiforov, V.: Detection of Abrupt Changes: Theory and Application. Prentice Hall Englewoods Cliffs, NJ (1993)

3. Edwards, C., Spurgeon, S.K., Patton, R.J.: Sliding mode observers for fault detection and isolation. Automatica 36(4)**,** 541-553 (2000)

4. Jian, T., Khorasani, K., Tafazoli, S.: Parameter estimation-based fault detection, isolation and recovery for nonlinear satellite models. IEEE Transactions on Control System Technology 16(4), 799-808 (2008)

5. Marzat, J., Walter, E., Damongeot, F., Piet-Lahanier, H.: Robust automatic tuning of diagnosis methods via an efficient use of costly simulations. In Proc. of the 16th IFAC Symposium on System Identification, Brussels, Belgium, 398-403 (2012)

6. Patton, R.J., Uppal, F.J., Simani, S., Polle, B.: Robust FDI applied to thruster faults of a satellite system. Control Engineering Practice 18(9), 1093-1109 (2009)

7. Sidi, M.J.: Spacecraft dynamics and control: a practical engineering approach. Cambridge University Press (1997)

8. Talebi, H.A., Khorasani, K. Tafazoli,S.: A recurrent neural-network-based sensor and actuator fault detection and isolation for nonlinear systems with application to the satellite's attitude control subsystem. IEEE Transactions on Neural Networks 20(1), 45-60 (2009)

9. D. Theilliol, D., Join, C., Zhang, Y.: Actuator Fault Tolerant Control Design Based On A Reconfigurable Reference Input. Journal of Applied Mathematics and computer Science 18(4), 553-560 (2008)

10. Varga, A.: Monitoring actuator failures for a large transport aircraft – the nominal case. In Proc. of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain, 627-632 (2009)

11. Wie, B.: Space Vehicle Dynamics and Control. AIAA Educational Series, Reston, Virginia (1997)

12. Williamson, W.R., Speyer, J.L., Dang, V.T., Sharp, J.: Fault Detection and Isolation for Deep Space Satellites. Journal of Guidance, Control and Dynamics 32(5), 1570-1584 (2009)

13. Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. Annual Reviews in Control 32(2), 229-252 (2008)