

## Chapter 3

# Experimental results of fault isolation approach for a 2-DOF helicopter

Manuel Alejandro Zuñiga<sup>1</sup>, Luis Alejandro Ramírez<sup>1</sup>, Gerardo Romero Galvan<sup>1</sup>, Efraín Alcorta García<sup>2</sup>, and Aldo J. Muñoz Vázquez<sup>3</sup>

**Abstract** Fault isolation increasingly becomes an important aspect for the feasibility and robustness of systems. The knowledge of a fault's presence offers important decision-making information related to the control of a system (i.e. information to keep a system in a healthy and functional state) and/or also for organizing maintenance. The case of a two degree of freedom (2-DOF) helicopter is studied. Practical aspects related to the fault modeling and implementation, in order to have the possibility of a detailed study of the used algorithm, are discussed. A previously developed model-based fault isolation algorithm is implemented by using specific hardware. A comparison between simulations and experimental results is also provided, and some directions for facing remaining challenges are also discussed.

### 3.1 Introduction

In recent years, fault diagnosis has been an important research topic, mainly because of its relationship with the safety and reliability of industrial processes. Fault diagnosis is useful for establishing maintenance programs for devices that are part of control systems and for developing fault-tolerant control schemes. In the last 50 years, different approaches for isolating faults in control systems have been proposed, as seen in Blanke et al. (2016), Shen et al. (2017), Varga (2017), Zhang et al. (2018), and Ding (2020). A failure in a control system is considered when a change in some parameters of the system occurs in such a way that it operates outside the tolerance margin for which it was designed. For this reason, it is important to establish mechanisms that allow determining when a fault occurs, which is known as the fault diagnosis and isolation problem.

One of the first analytical redundancy algorithms for fault diagnosis is the so-called model-based approach, i.e. they are based on mathematical models, which can be consulted in Blanke et al. (2016) and Ding (2020); here, the mathematical model plays a central role in the design of fault detection and isolation algorithms. Other approaches that have been recently proposed appear in Patan (2019), where neural networks are used primarily to determine fault-tolerant control algorithms; Concepts based on

---

<sup>1</sup>Universidad Autónoma de Tamaulipas, Unidad Académica Multidisciplinaria Reynosa Rodhe, Mexico, e-mail: manuel.zuniga@docentes.uat.edu.mx, laramirez@docentes.uat.edu.mx and gromero@uat.edu.mx

<sup>2</sup>Universidad Autónoma de Nuevo León, Facultad de Ingeniería Mecánica y Eléctrica, Mexico, e-mail: efrain.alcortagr@uanl.edu.mx

<sup>3</sup>Texas A&M University, Department of Multidisciplinay Engineering, McAllen, TX 78504, USA, e-mail: aldo.munoz.vazquez@gmail.com

artificial intelligence have also been used to solve this problem, see Korbicz et al. (2004). Furthermore, artificial intelligence and data-driven approaches have been employed to solve this problem. See Korbicz et al. (2004) and Mansouri et al. (2020), as well as in Ding (2014) who includes some model-inspired approaches. Fault diagnosis of hybrid dynamic and complex systems has been considered in Sayed-Mouchaweh (2018). Furthermore an adaptive techniques for fault diagnosis are discussed in Shen et al. (2017), and Wang et al. (2020) consider iterative learning approaches for fault diagnosis. The diagnosis of fractional-order systems is studied in Martinez-Guerra et al. (2021), and fault diagnosis in switched systems is revised in Du et al. (2021). Among the above mentioned books considering principally theoretical results, there are also references that consider specific applications, such as Zolghadri et al. (2014) for fault diagnosis, fault-tolerant control, and the guidance of aerospace vehicles. Other aspects such as the fault detection of wind turbine systems have been considered in Reza Karimi (2018). Induction motors fault diagnosis is discussed in Karmakar et al. (2016). Model-based condition monitoring, actuator, sensors, machinery, plants, and drives are considered in Isermann (2011).

This work presents an analysis and design of a fault isolation scheme for a Quanser® 2-DOF Helicopter. Mainly, a Hamiltonian model of the system previously mentioned is derived. Faults in the system are also modeled as well as implementation methods (with the results of this study). A fault isolation scheme is derived using nonlinear decoupling and observer-based residual generation, as in Ramírez et al. (2020). The implementation of the fault isolation algorithms, as well as the experimental results, are also discussed. Some practical experiences related to the implementation are presented.

The rest of this work is organized as follows: the next section shows some preliminary results that will be used later in the paper. The system under consideration is presented in Section 3, and details about the design of the fault isolation scheme are considered in Section 4. The experimental results are presented in Section 5, and, finally, some conclusions are summarized in Section 6.

## 3.2 Preliminaries

### 3.2.1 Generalized Hamiltonian systems

A class of nonlinear systems admits a representation in a generalized Hamiltonian form, as described by Rodriguez-Alfaro et al. (2015), Sira Ramírez and Cruz Hernández (2001) and Van der Schaft (2017). It is given by

$$\begin{aligned} \dot{x} &= [J(x) + S(x)] \frac{\partial H(x)}{\partial x} + F(x) + Gu \\ y &= C \frac{\partial H(x)}{\partial x} \end{aligned} \quad (3.1)$$

with the state vector  $x \in \mathfrak{R}^n$ , the input vector  $u \in \mathfrak{R}^m$ ,  $y \in \mathfrak{R}^p$  is the output vector,  $G \in \mathfrak{R}^{n \times m}$  being a constant matrix,  $F \in \mathfrak{R}^n$  a vector of nonlinear functions and  $J \in \mathfrak{R}^{n \times n}$  being a skew-symmetric interconnection matrix ( $J(x) = -J^T(x)$ ) associated with a simple bilinear form, which represents the conservative part of the system.  $S(x) \in \mathfrak{R}^{n \times n}$  is a symmetric matrix ( $S(x) = S^T(x)$ ),  $C \in \mathfrak{R}^{p \times n}$  is an output constant matrix.  $H(x)$  is a smooth vector-valued function of and real values, and it can be understood as the

generalized energy defined as

$$H(x) = \frac{1}{2}x^T Mx \quad (3.2)$$

where  $M$  is a constant matrix, symmetric and positive definite, and  $\partial H(x)/\partial x$  represents the gradient of  $H(x)$ . Finally,  $F(x)$  is a vector field that contains the non-linear parts of the system.

### 3.2.2 Fault isolation schema

The fault isolation strategy utilized in this paper was presented in Ramírez et al. (2020); it is basically a bank of nonlinear observer-based residual generators. Each residual generator is designed for a decoupled subsystem so that each subsystem is associated with a set of faults. Moreover, each subsystem is obtained by taking advantage of the specific model structure of the 2-DOF helicopter.

In addition, the decoupling of the effect of some specific faults could be achieved by using the output measurements, which are linear with respect to the state. This will be made clear later. Roughly speaking, the residuals can be represented as shown in Fig. 3.1.

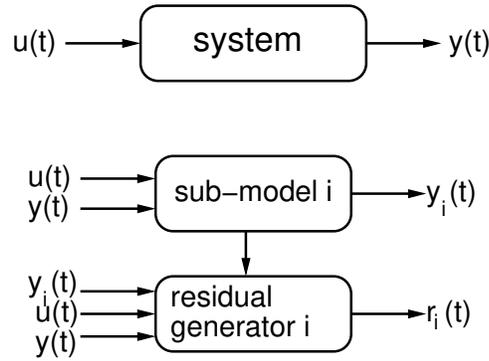


Fig. 3.1: Schematic of residuals

### 3.3 Helicopter with two degrees of freedom

The system used in this work is the Quanser® 2-DOF helicopter, which is a lab test prototype standing on a fixed pedestal (that pivots) and consists of two motors: one on the front for the lift movement and one rear or tail motor for the yaw movement. See Fig. 3.2.

The system model is represented by means of the following nonlinear equation, in vector form:

$$D(q)\ddot{q} + N(q, \dot{q}) + g_v(q) = \tau, \quad (3.3)$$

where  $\tau \in \mathfrak{R}^2$  represents the input vector,  $g_v(q) \in \mathfrak{R}^2$  is the vector of gravitational pairs,  $D(q) \in \mathfrak{R}^{2 \times 2}$  corresponds to the inertial matrix, which is symmetric and positive definite,  $N(q, \dot{q}) \in \mathfrak{R}^{2 \times 2}$  models the

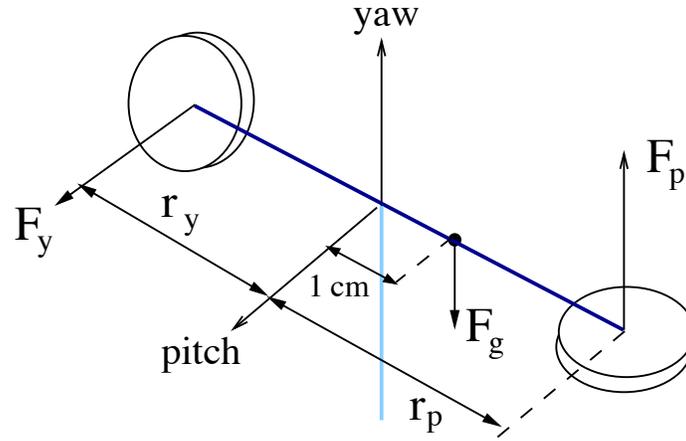


Fig. 3.2: Schematic of the Quanser® 2-DOF helicopter

velocity quadratic effects generated by centrifugal and Coriolis forces. For the Quanser® system, define the generalized coordinate vector  $q = [\phi, \psi]^T$ , where  $\phi$  and  $\psi$  are the pitch and yaw angles. Consider  $\tau = [U_p, U_y]^T$ , and

$$D(q) = \begin{bmatrix} m_1 l_1^2 + m_2 l_2^2 & 0 \\ 0 & m_1 l_1^2 \cos^2(\phi) + m_2 l_2^2 \cos^2(\phi) \end{bmatrix},$$

$$g_v(q) = \begin{bmatrix} m_1 g l_1 \cos(\phi) - m_2 g l_2 \cos(\phi) \\ 0 \end{bmatrix}.$$

### 3.3.1 Hamiltonian representation

Hamiltonian representation brings a structure to a model of a dynamic system and this will be used later to design a fault detection and isolation scheme.

A way to obtain the Hamiltonian representation from the Euler-Lagrange representation (3.3) (as in Van der Schaft (2017), for example) is by computing the generalized moment  $p(t)$ , defined by

$$p = D(q)\dot{q} \quad (3.4)$$

and the Hamiltonian function

$$\mathcal{H}(q, p) = \frac{1}{2} p^T D^{-1}(q) p + U(q), \quad (3.5)$$

where  $U(q)$  represents the potential energy of the system. Thus, the system's model (3.3) can be alternatively written as

$$\dot{q} = \frac{\partial \mathcal{H}(q, p)}{\partial p} = D^{-1}(q)p, \quad (3.6)$$

$$\dot{p} = -\frac{\partial \mathcal{H}(q, p)}{\partial q} + \tau = -\frac{\partial}{\partial q} \left( \frac{1}{2} p^T D^{-1}(q)p \right) - \frac{\partial}{\partial q} U(q) + \tau, \quad (3.7)$$

where  $\frac{\partial}{\partial q} U(q) = g_v(q)$ .

The system equations that represent (3.3) are the following:

$$\dot{x}_1 = \frac{x_3}{l^2(m_1 + m_2)}, \quad (3.8)$$

$$\dot{x}_2 = \frac{x_4}{l^2 \cos^2(x_1)(m_1 + m_2)}, \quad (3.9)$$

$$\dot{x}_3 = \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1 + m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p, \quad (3.10)$$

$$\dot{x}_4 = U_y \quad (3.11)$$

where the system state vector is given by  $x^T = [x_1 \ x_2 \ x_3 \ x_4]$  with  $x_1 = \phi$ ,  $x_2 = \psi$ ,  $x_3 = \dot{\phi}$  and  $x_4 = \dot{\psi}$ . The control law  $\tau = [U_p, U_y]^T$  is a proportional derivative (PD) type controller, and it was tuned using a procedure presented in Romero et al. (2012). The variables and parameters used in the previous equations are as follows:  $x_1(\phi)$  is the pitch angle;  $x_2(\psi)$  is the yaw angle;  $x_3(\dot{\phi})$  corresponds to the pitch angular velocity;  $x_4(\dot{\psi})$  represents the pitch angular velocity;  $x_4(\dot{\psi})$  gives the yaw angular velocity;  $U_p$  is the pitch control input;  $U_y$  corresponds to the yaw control input;  $m_1$  represents the mass of lift motor;  $m_2$  is the mass of the tail motor;  $l_1$  corresponds to the length  $m_1$  to center of mass;  $l_2$  is the length  $m_2$  to center of mass and  $g$  represents the gravity.

The following generalized Hamiltonian function is defined:  $H = \frac{1}{2} x^T M x$ , with  $M = I_{4 \times 4}$ .  $H$  is the basis of the generalized Hamiltonian representation

$$\dot{x} = J(x) \frac{\partial H(x)}{\partial x} + S(x) \frac{\partial H(x)}{\partial x} + \begin{bmatrix} \frac{x_3}{l^2(m_1+m_2)} \\ \frac{x_4}{l^2 \cos^2(x_1)(m_1+m_2)} \\ \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1+m_2)} - gl \cos(x_1)(m_1 - m_2) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} U_p \\ U_y \end{bmatrix}}_u, \quad (3.12)$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \frac{\partial H(x)}{\partial x}, \quad (3.13)$$

where,  $\frac{\partial H}{\partial x} = x$ ,  $J(x) = [0] \in \mathbb{R}^{4 \times 4}$ ,  $S(x) = [0] \in \mathbb{R}^{4 \times 4}$ .

### 3.3.2 Fault Modeling

Four possible faults are studied in this work: one for each sensor and one for each actuator. In general, faults could be additive or multiplicative; however, in this paper the focus is on the case of additive faults, which are modeled as exogenous inputs to the system. Note that faults can be also incipient, abrupt or

intermittent, depending on the form in which they are manifested. See for example Chen and Patton (1999), Ding (2013) and Isermann (2011).

### 3.3.2.1 Additive faults' representation

As mentioned above, additive faults are modeled with additive terms in the actuators channels  $f_{ai}$ , as well as in the sensors channels  $f_{si}$ , with  $i = 1, 2, 3$ . The representation of the system's model with faults (sensor and actuator) is given as follows:

$$\dot{x} = J(x)\frac{\partial H(x)}{\partial x} + S(x)\frac{\partial H(x)}{\partial x} + F(x) + Gu + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{E_f} \begin{bmatrix} f_{a1} \\ f_{a2} \end{bmatrix}, \quad (3.14)$$

$$y = C\frac{\partial H(x)}{\partial x} + \begin{bmatrix} f_{s1} \\ f_{s2} \end{bmatrix}. \quad (3.15)$$

### 3.3.2.2 Fault decoupling

Sensor faults

In order to obtain two sub-models where each of them is associated with only one sensor fault, the following strategy is used. Firstly, the output related to sensor one (output one) is considered; here, the corresponding differential equation (3.8) of the states required to integrate output one is considered as part of the sub-model one. Additionally, the differential equations of the states appearing in the previous differential equations, i.e. equations (3.9), (3.10) and (3.11), should be included in the sub-model as well.

Secondly, referring to sub-model two, output two (related to sensor two) is considered. The corresponding differential equations of the states required to integrate output two are considered as part of the sub-model two. As in the first case, the differential equations of the states appearing in the previous differential equations should also be included in the sub-model.

**Sub-model one.** For the first sensor, only the state  $x_1$  is involved, so the differential equation for  $\dot{x}_1$  is required first. Because,  $x_3$  appears in the equation of  $\dot{x}_1$ , however, the differential equation for  $\dot{x}_3$  should also be included in the sub-model. Note that  $x_4$  is present in the differential equation of  $\dot{x}_3$ , so the differential equation for  $\dot{x}_4$  is also required in sub-model one:

$$\dot{x}_1 = \frac{x_3}{l^2(m_1 + m_2)}, \quad (3.16)$$

$$\dot{x}_3 = \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1 + m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p + f_{a1}, \quad (3.17)$$

$$\dot{x}_4 = U_y + f_{a2}, \quad (3.18)$$

$$y_{s1} = x_1 + f_{s1} \quad (3.19)$$

Note that in sub-model one, the fault of sensor two ( $f_{s2}$ ) is not affecting, i.e. it is not present in the submodel one.

**Sub-model two.** For the second subsystem, only the second output is considered. The state variable involved is  $x_2$ , so  $\dot{x}_2$  should be included. In the differential equation of  $\dot{x}_2$ ,  $x_1$  and  $x_4$  are also present; therefore,  $\dot{x}_1$  and  $\dot{x}_4$  are included. Finally,  $x_3$  is also present, so  $\dot{x}_3$  is also included:

$$\dot{x}_1 = \frac{x_3}{l^2(m_1 + m_2)}, \quad (3.20)$$

$$\dot{x}_2 = \frac{x_4}{l^2 \cos^2(x_1)(m_1 + m_2)}, \quad (3.21)$$

$$\dot{x}_3 = \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1 + m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p + f_{a1}, \quad (3.22)$$

$$\dot{x}_4 = U_y + f_{a2}, \quad (3.23)$$

$$y_{s2} = x_2 + f_{s2} \quad (3.24)$$

Note that in the sub-model two, the fault of sensor one ( $f_{s1}$ ) is not affecting, i.e. it is not present in the submodel two.

#### Actuator faults

Similar to the case of sensor faults, we obtained the decoupled subsystems that are sensitive only to one particular fault. Thus, the corresponding residual is generated from an observer based on output feedback.

**Sub-model three.** The first actuator fault ( $f_{a1}$ ) is connected to the differential equation of the state variable  $x_3$  of  $U_p$  since  $x_1$  is known and  $x_2$  is assumed to be free of fault.

$$\dot{x}_1 = \frac{x_3}{l^2(m_1 + m_2)}, \quad (3.25)$$

$$\dot{x}_3 = \frac{-(\dot{x}_2 l^2 \cos^2(x_1)(m_1 + m_2))^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1 + m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p + f_{a1}, \quad (3.26)$$

$$y_1 = x_1 + f_{s1}. \quad (3.27)$$

Note that  $x_4$  is not available, but it can be obtained from the dynamics of the system as

$$x_4 = \dot{x}_2 l^2 \cos^2(x_1)(m_1 + m_2), \quad (3.28)$$

and  $\dot{x}_2$  is obtained through numerical differentiation from the measurement:  $y_2 = x_2 + f_{s2}$ .

**Sub-model four.** The differential equation for  $\dot{x}_4$  is now considered. Note that no additional state variable is strictly required, however, in order to connect with the output the differential equation because  $\dot{x}_2$  is also included in the model together with the second output:

$$\dot{x}_2 = \frac{x_4}{l^2 \cos^2(x_1)(m_1 + m_2)} = \frac{x_4}{l^2 \cos^2(y_1 - f_{s1})(m_1 + m_2)}, \quad (3.29)$$

$$\dot{x}_4 = U_y + f_{a2}, \quad (3.30)$$

$$y_2 = x_2 + f_{s2}. \quad (3.31)$$

in addition  $x_1$  is obtained from the first output equation as:  $x_1 = y_1 - f_{s1}$

*Remark 1.* Note that for fault decoupling an inspection procedure is used instead of a nonlinear transformation, in other words, for sensor  $i$ , the output is selected (omitting all other outputs) as well as all the differential equations associated with the states that explicitly appear in the selected output and the ones required to complete the states present in the differential equations selected. In Table 3.1 a summary of the sensitivity of each of the different faults is presented, where  $\checkmark$  means that the sub-model is sensitive to the corresponding fault.

The fault incidence matrix is as follows:

Table 3.1: Sensitivity of sub-models with respect to faults

sub-models	$f_{s1}$	$f_{s2}$	$f_{a1}$	$f_{a2}$
1	$\checkmark$	0	$\checkmark$	$\checkmark$
2	0	$\checkmark$	$\checkmark$	$\checkmark$
3	$\checkmark$	$\checkmark$	$\checkmark$	0
4	$\checkmark$	$\checkmark$	0	$\checkmark$

where  $\checkmark$  means that the corresponding sub-model is affected (at least theoretically) by the respective fault. In contrast, 0 means that the sub-model is decoupled from the respective fault.

## 3.4 Fault isolation for a 2-DOF Helicopter

### 3.4.1 Residual Generator Design

The observer-based residuals are built following the method presented in Ramírez et al. (2020). It takes advantage of the Hamiltonian structure of the model, and it is rewritten here for completeness:

$$\dot{x} = [J(y) + S(y)] \frac{\partial H(x)}{\partial x} + F(x) + Gu, \quad (3.32)$$

$$y = C \frac{\partial H(x)}{\partial x}. \quad (3.33)$$

The state  $x$  of the nonlinear system (3.1) is estimated by the system

$$\dot{\hat{x}} = [J(y) + S(y)] \frac{\partial H(\hat{x})}{\partial \hat{x}} + F(\hat{x}) + Gu + K(y - \eta), \quad (3.34)$$

$$\eta = C \frac{\partial H(\hat{x})}{\partial \hat{x}}, \quad (3.35)$$

if the pair  $(C, S)$  is observable or at least detectable and the matrix

$$\Xi = M^T \left[ S - \frac{1}{2}(KC + C^T K^T) \right] M + \Pi \quad (3.36)$$

with  $\Pi = \frac{1}{2} \left[ M \frac{\partial F(x)}{\partial x} + \frac{\partial F(x)}{\partial x}^T M^T \right]$  is negative definite.

For the construction of the bank of residual generators, an observer for each sub-model should be designed. Considering the sub-model one, which has been obtained to have the effect of the sensor of fault one ( $f_{s1}$ ), and the analysis made, the actuator faults  $f_{a1}$  and  $f_{a2}$  also affect sub-model one:

$$\dot{x}_1 = \frac{x_3}{l^2}(m_1 + m_2), \quad (3.37)$$

$$\dot{x}_3 = \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1 + m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p, \quad (3.38)$$

$$\dot{x}_4 = U_y + f_{a2}, \quad (3.39)$$

$$y_{s1} = x_1 + f_{s1}. \quad (3.40)$$

leads to its generalized Hamiltonian representation, with

$$\frac{\partial H}{\partial x} = [x_1 \quad x_3 \quad x_4]^T. \quad (3.41)$$

The measurement of the output variables was performed by two rotary optical encoders: one placed at the base of the platform, which allows the yaw angle to be measured  $\psi$ , and the other placed in the center of the platform, which measures the pitch angle  $\phi$ . See Fig 3.3.

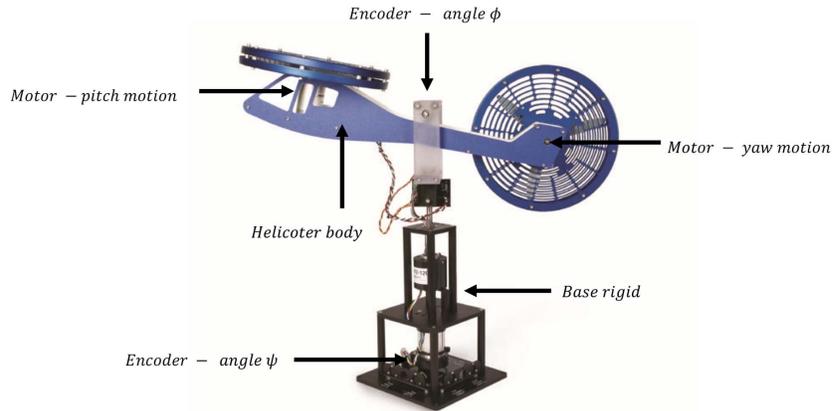


Fig. 3.3: Helicopter structure taken from Quanser Inc. (2012)

The parameters of the previous model are the following:

$$J(x) = [0] \in \mathbb{R}^{3 \times 3}, \quad S(x) = [0] \in \mathbb{R}^{3 \times 3}, \quad F(x) = \begin{bmatrix} \frac{x_3}{l^2(m_1+m_2)} \\ \frac{-x_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1+m_2)} - gl \cos(x_1)(m_1 - m_2) \\ 0 \end{bmatrix},$$

$$G = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C = [1 \ 0 \ 0].$$

The pair  $(C, S)$  is detectable according to Definition 2.1 in Sira Ramírez and Cruz Hernández (2001) since the matrix  $\begin{bmatrix} C \\ \lambda I_{3 \times 3} - S \end{bmatrix}$  is of full rank for any value of  $\lambda$  in  $\mathbb{Z}_{<0} = \{\bar{\lambda} \in \mathbb{Z} : \Re\{\bar{\lambda}\} < 0\}$ . Thus, the residual is defined from the observer:

$$\begin{aligned} \dot{\hat{x}}_1 &= \frac{\hat{x}_3}{l^2(m_1+m_2)} + L_1(y_1 - \hat{x}_1), \\ \dot{\hat{x}}_3 &= \frac{-\hat{x}_4^2 \sin(x_1)}{l^2 \cos^3(x_1)(m_1+m_2)} - gl \cos(x_1)(m_1 - m_2) + U_p, \\ \dot{\hat{x}}_4 &= U_y, \\ \hat{y}_{s1} &= \hat{x}_1. \end{aligned}$$

Note that the residual is designed from a copy of the original subsystem and the correction factor  $L_1(y_1 - \hat{y}_1)$ .

For the fault in the second sensor ( $f_{s2}$ ) and those in the two actuators ( $f_{a1}$ ) and ( $f_{a2}$ ), the decoupling is similar to the procedure utilized in the fault ( $f_{s1}$ ).

In Table 1, we can see that if single faults are occurring, they can be isolated. The four residuals are obtained assuming that only one fault at a time is present and not all simultaneously. In addition, a residual is obtained considering a derivative in time, so obtaining this derivative could be a difficult task in the presence of noise. It is important to notice that the generation of residuals based on observers constitutes a reliable option for reducing noise effects because of the inherent filtering characteristic of the observer; see Rodriguez-Alfaro et al. (2015).

### 3.5 Experimental results

The 2-DOF experimental platform consists of a helicopter's body mounted on a rigid metal base. It has two motors mounted perpendicularly to each other. This emulates the typical helicopter configuration with a main rotor, which generates the pitch motion, and a tail rotor that produces the yaw motion. DC motors work with a nominal voltage of 12 V. The platform parameters are shown in Table 3.2.

The measurement of the output variables was carried out by two rotary optical encoders: one placed at the base of the platform, which allows the yaw angle to be measured  $\psi$ , and the other placed in the center of the platform, which measures the pitch angle  $\phi$  (3.3).

Table 3.2: The platform parameters

Parameter	Description	Value
$m_1$	Mass of lift motor	$0.014kg$
$m_2$	Mass of tail motor	$0.232kg$
$l_1$	Length $m_1$ to center of mass	$0.203m$
$l_2$	Length $m_2$ to center of mass	$0.203m$
$g$	Gravity	$9.8m/s^2$

To implement the algorithm, the Simulink<sup>®</sup> tool from MATLAB<sup>®</sup> was used. Data acquisition was performed using an Arduino Mega 2560, which consists of a 54 input/output pins, including 14 pins with the ability to work as pulse width modulation and 16 pins as analog inputs, a microprocessor, a 16 MHz crystal, and a 256K flash memory, with a working range between 7 V and 12 V. In the power part, a SparkFun Monster Moto Shield was used, with a capacity of 16 V, and a current of 30 A, 20 kHz of PWM frequency.

It is important to mention that the faults produced in each of the physical tests were in a proportion of 50% with respect to the output, and they were presented as abrupt faults.

#### *Introduced sensor faults*

They are considered abrupt failures in the sensors; they appear suddenly and affect the measured signal of the encoders. As a result, the signal used to read the pitch and yaw angles, which is later fed back to the control algorithm, presents deviations between estimated and the current (real) angle of the helicopter position.

To produce a failure in the sensors, we considered adding an external signal to the sensor's signal that feeds the algorithm. The signal is formed virtually from a block of noise-limited according to a percentage of the failure with respect to the output.

#### *Introduced actuator faults*

Helicopter motors are considered actuators that convert electrical energy into rotary motion. Propellers generate thrust force that produces displacement in pitch and yaw, respectively. Because of the aerodynamic profile of the propellers, when rotating, a difference in speed is obtained on each side of the blade between the fluid on one side and the other. Therefore, the difference in speeds leads to a difference in pressure, and, thus, a force that is perpendicular to the plane of rotation is generated, which is known as the propulsion force. Once the helicopter propeller begins to rotate, it picks up the air and is an accelerator. The speed of the air behind the disk, multiplied by the mass of displaced air, gives us the thrust force. Based on this idea and in order to cause a decrease in the thrust force of the helicopter, different ways of blocking the air (like covers) surrounding the helicopter were designed that correspond to 50% of the area of the rotating disk, which, when placed on the propellers, reduced the airflow, causing the

helicopter to lose thrust and lift. These covers were placed abruptly in order to have the same effect as a partial actuator failure.

### 3.5.1 Fault scenarios and experimental results

In order to show the effectiveness of the designed residuals, four cases are presented, one for each considered fault. For each case, the four residuals obtained from the experimental platform are shown.

#### Fault in sensor one

A 50% change in the sensor signal of the output one is considered, i.e., of the measured angle  $\phi$ . The change occurs 15 seconds after the initial time.

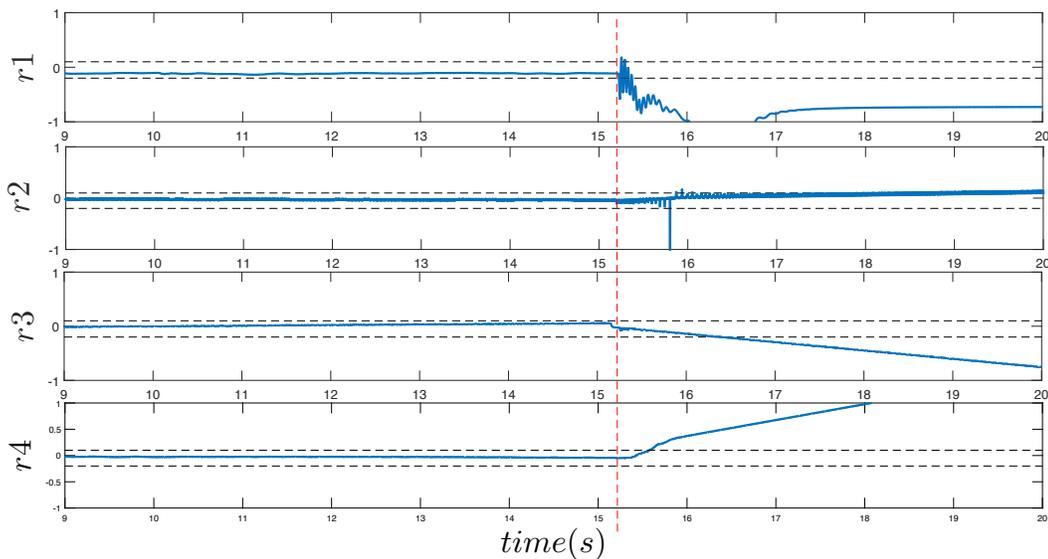


Fig. 3.4: Residuals' response to sensor fault one ( $f_{s1}$ )

#### Fault in sensor two

A 50% change in the sensor signal of output two is considered, i.e., of the measured angle  $\psi$ . The change occurs 15 seconds after the initial time.

#### Fault in actuator one

An approximated 50% change in the first actuator signal is considered. The change occurs 15 seconds after the initial time approximately.

#### Fault in actuator two

An approximated 50% change in the second actuator signal is considered. The change occurs approximately 15 seconds after the initial time.

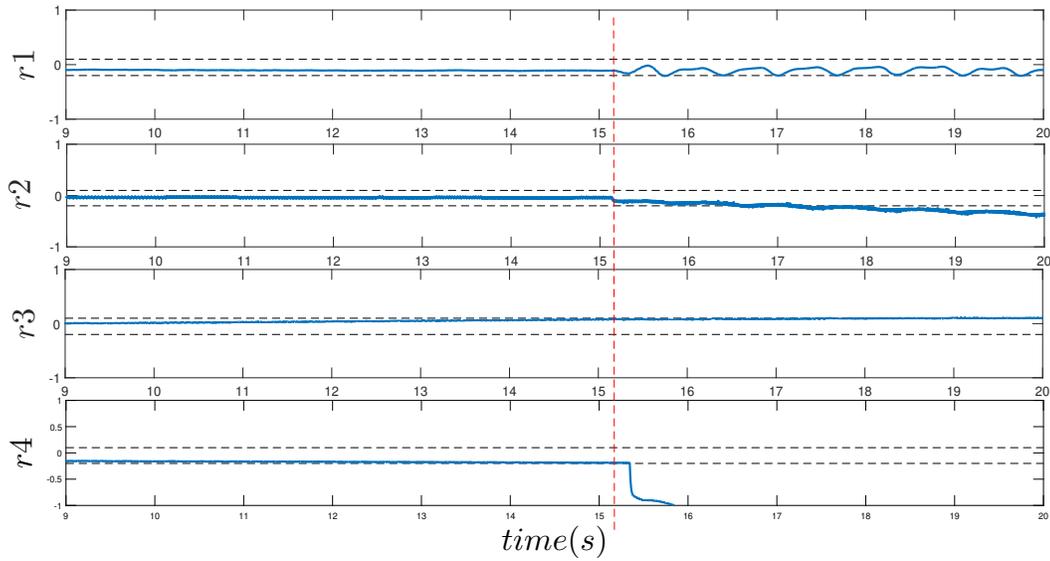


Fig. 3.5: Residuals' response to sensor fault two ( $f_{s2}$ )

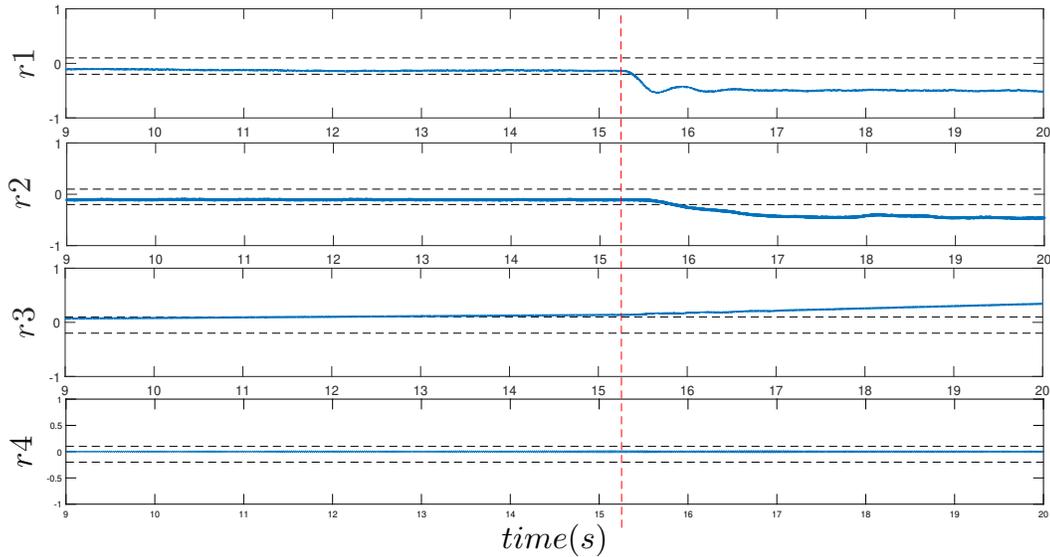


Fig. 3.6: Residuals' responses to an actuator one fault ( $f_{a1}$ )

### 3.6 Conclusion

Model-based fault isolation algorithms represent an interesting alternative for diagnosis in many processes. Fault isolation should be considered from the beginning of the design processes in order to obtain a better result. Because the considered faults correspond to the sensor and actuator, a simple procedure was used in order to obtain dynamic redundant relations that can be used for the residual design. Considering a class of port-Hamiltonian nonlinear systems, a systematic observer-based procedure for residual design was considered. The designed residuals were tested in a simulation and afterwards were imple-

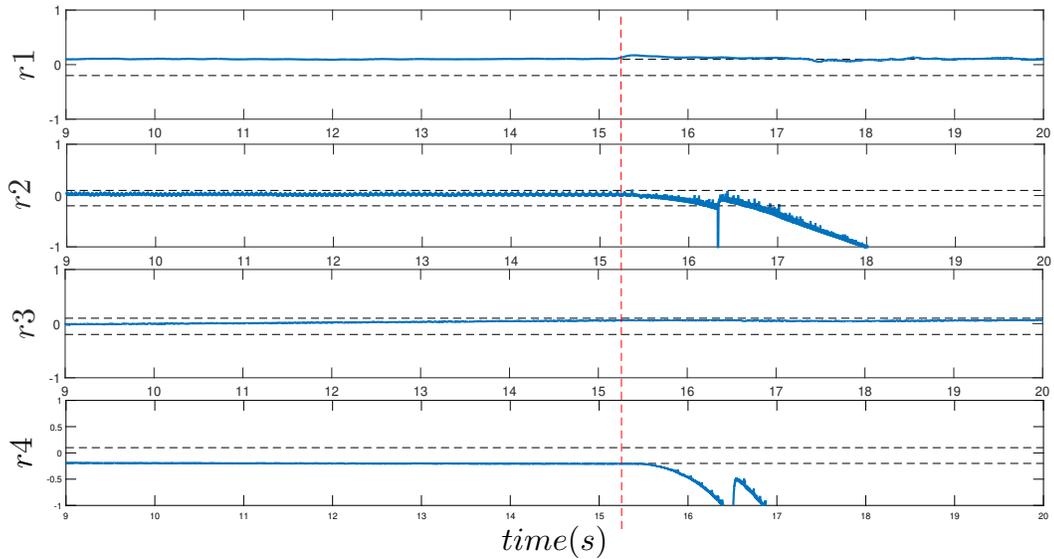


Fig. 3.7: Residuals' responses to an actuator two fault ( $f_{a2}$ )

mented to be tested in a real-time embedded system. Fault isolation was obtained for the four single faults considered, in which one case also required a specific residual evaluation algorithm (how the sensor fault two, residual one, which should have been zero, but it had some reaction). An important aspect was how the fault was implemented because it allowed us to make a repetitive experiment. Some topics that could be part of future work include seeking to reduce the thresholds for fault detection and working, in this way, with small magnitude faults. On the other hand, it is essential to reduce fault detection time and isolate coupled faults. Finally, it is also of great interest to propose control strategies that allow the system to be tolerant when a fault occurs.

## References

- Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M. (2016). *Diagnosis and Fault-Tolerant Control*. Springer.
- Chen, J. and Patton, R. J. (1999). *Robust model based fault diagnosis for dynamic systems*. Kluwer Academic Publishers Group.
- Ding, S. X. (2013). *Model-based fault diagnosis techniques 2nd edition*. Springer, 2nd edition.
- Ding, S. X. (2014). *Data-driven Design of Fault Diagnosis and Fault-tolerant Control Systems*. Advances in Industrial Control. Springer, London.
- Ding, S. X. (2020). *Advanced methods for fault diagnosis and fault-tolerant control*. Springer.
- Du, D., Xu, S., and Cocquempot, V. (2021). *Observer-Based Fault Diagnosis and Fault-Tolerant Control for Switched Systems*. Springer.
- Isermann, R. (2011). *Fault-Diagnosis Applications*. Springer.
- Karmakar, S., Chattopadhyay, S., Mitra, M., and Sengupta, S. (2016). *Induction Motor Fault Diagnosis*. Springer.
- Korbicz, J., Koscielny, J. M., Kowalczyk, Z., Cholewa, W., and (Eds.) (2004). *Fault Diagnosis*. Springer.
- Mansouri, M., Harkat, M.-F., Nounou, H. N., and Nounou, M. N. (2020). *Data-driven and model-based methods for fault detection and diagnosis*. Elsevier.
- Martinez-Guerra, R., Meléndez-Vázquez, F., and Trejo-Zúñiga, I. (2021). *Fault-tolerant control and diagnosis for integer and fractional-order systems*. Springer.
- Quanser Inc. (2012). *Laboratory Guide, 2 DOF Helicopter Experiment for MATLAB®/Simulink® Users*. Quanser Inc., Markham, Ontario, Canada, 1. edition.
- Patan, K. (2019). *Robust and Fault-Tolerant Control: Neural-Network-Based Solutions*. Springer.
- Ramírez, L. A., Zuñiga, M. A., Romero, G., Alcorta-García, E., and Muñoz-Vázquez, A. J. (2020). Fault diagnosis for a class of robotic systems with application to 2-dof helicopter. *Applied Sciences*, 10(23).
- Reza Karimi, H. (2018). *Structural control and fault detection of wind turbine systems*. The Institution of Engineering and Technology.
- Rodriguez-Alfaro, L. H., Alcorta-Garcia, E., Lara, D., and Romero, G. (2015). Hamiltonian approach to fault isolation in a planar vertical take-off and landing aircraft model. *Int. J. Appl. Math. Comput. Sci.*, 25:65–76.
- Romero, G., Alcorta-García, E., Lara, D., Pérez, I., Betancourt, B., and Ocampo, H. (2012). New method for tuning robust controllers applied to robot manipulators. *International Journal of Advanced Robotic Systems*.

- Sayed-Mouchaweh, M., editor (2018). *Fault diagnosis of hybrid dynamic and complex systems*. Springer.
- Shen, Q., Jiang, B., and Shi, P. (2017). *Fault Diagnosis and Fault-Tolerant Control Based on Adaptive Control Approach*. Springer.
- Sira Ramírez, H. and Cruz Hernández, C. (2001). Synchronization of chaotic systems: a generalized hamiltonian systems approach. *International journal of bifurcation and chaos*, 11:1381–1395.
- Van der Schaft, A. (2017). *L2-gain and passivity techniques in nonlinear control*. Springer.
- Varga, A. (2017). *Solving Fault Diagnosis Problems*. Springer.
- Wang, L., Zhang, R., and Gao, F. (2020). *Iterative Learning Stabilization and Fault-Tolerant Control for Batch Processes*. Springer.
- Zhang, K., Jiang, B., Shi, P., and Cocquempot, V. (2018). *Observer-Based Fault Estimation Techniques*. Springer.
- Zolghadri, A., Henry, D., Cieslak, J., Efimov, D., and Goupil, P. (2014). *Fault Diagnosis and Fault-Tolerant Control and Guidance for Aerospace Vehicles, From Theory to Application*. Springer.