



# Algorithms and architectures for control and diagnosis of flight critical systems

Alexandre Bobrinskoy

## ► To cite this version:

Alexandre Bobrinskoy. Algorithms and architectures for control and diagnosis of flight critical systems. Automatic. Université de Bordeaux, 2015. English. NNT : 2015BORD0007 . tel-01249590

**HAL Id: tel-01249590**

**<https://theses.hal.science/tel-01249590>**

Submitted on 4 Jan 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE PRÉSENTÉE  
POUR OBTENIR LE GRADE DE  
**DOCTEUR DE**  
**L'UNIVERSITÉ DE BORDEAUX**

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR  
SPÉCIALITÉ : AUTOMATIQUE

par Alexandre BOBRINSKOY

**ALGORITHMES ET ARCHITECTURES POUR  
LA COMMANDE ET LE DIAGNOSTIC DE  
SYSTÈMES CRITIQUES DE VOL**

---

**ALGORITHMS AND ARCHITECTURES FOR  
CONTROL AND DIAGNOSIS OF FLIGHT  
CRITICAL SYSTEMS**

Sous la direction de : Franck CAZAURANG  
(co-encadrant : Bruno BLUTEAU)

Soutenue le : 29 Janvier 2015

Membres du jury :

M. GLUMINEAU, Alain	Professeur à l'École Centrale de Nantes	Rapporteur
M. LAROCHE, Edouard	Professeur à l'Université de Strasbourg	Rapporteur
M. GATTI, Marc	THALES Avionics	Examineur
M. GUÉRINEAU, Olivier	THALES Systèmes Aéroportés	Examineur
M. LAVIGNE, Loïc	Maître de Conférences à l'Université de Bordeaux	Examineur
M. MARÉ, Jean-Charles	Professeur à l'I.C.A., I.N.S.A. de Toulouse	Examineur
M. CAZAURANG, Franck	Professeur à l'Université de Bordeaux	Directeur de thèse
M. BLUTEAU, Bruno	Docteur au Laboratoire ESTIA Recherche	Co-encadrant



*A ma famille,  
A mes amis.*



# Remerciements

Le travail présenté dans cette thèse est le fruit d'une collaboration entre le Laboratoire d'Intégration du Matériau au Système (IMS) de l'Université de Bordeaux (équipe Flatness and Fault Tolerance Guidance (FFTG)), la société THALES Systèmes Aéroportés (service Centre Compétences Calculateur) à Pessac et le Laboratoire ESTIA-Recherche à Bidart.

Que Monsieur Alain Glumineau, Professeur à l'Institut de Recherche en Communications et Cybernétique de l'École Centrale de Nantes, et Monsieur Edouard Laroche, Professeur au Laboratoire des sciences de l'Ingénieur, de l'Informatique et de l'Imagerie de l'Université de Strasbourg trouvent ici l'expression de ma plus profonde gratitude, de m'avoir fait l'honneur de rapporter ce travail.

Je tiens également à remercier chaleureusement mon directeur de thèse, Monsieur Franck Cazaurang, Professeur à l'Université de Bordeaux ainsi que mon co-encadrant de thèse, Monsieur Bruno Bluteau, Professeur agrégé au laboratoire ESTIA-Recherche, pour m'avoir encadré et encouragé tout au long de mon travail. Je leur témoigne toute ma reconnaissance pour les fructueuses discussions que nous avons eues, leur amitié ainsi que leur soutien durant les phases d'étude et de réalisation du banc d'essai.

Je souhaite aussi remercier vivement Monsieur Olivier Guérineau, responsable d'Études Amont au sein de Thales Systèmes Aéroportés ainsi que Monsieur Marc Gatti, Directeur Technique au sein de Thales Avionics, pour la qualité de leur encadrement et de m'avoir transmis leur passion pour l'aéronautique.

Je remercie Monsieur Loïc Lavigne, Maître de conférences à l'Université de Bordeaux, pour son amitié et son soutien continu durant ce travail. Je suis très honoré de sa présence en tant que membre du jury.

Je tiens à exprimer également mes remerciements à Monsieur Jean-Charles Maré, Professeur à l'Institut National des Sciences Appliquées de Toulouse, pour la qualité de son accueil lors des différentes manifestations scientifiques, ainsi que pour ses explications fascinantes sur les technologies d'actionneurs aéronautiques. Qu'il trouve ici l'expression de mon amitié et de ma gratitude pour sa participation au jury.

Enfin je remercie tous mes collègues et amis pour leur joie et bonne humeur souvent très contagieuse. Merci à Romain, Mohcine, César, Aurélie, Enguerrand, Christophe, Marine, Marie, Hadrien, Antonin, Franck et Adrian pour ces moments inoubliables.

Je ne pourrais clore ces remerciements sans une pensée toute particulière à Laura, pour son soutien et sa patience durant ces trois années ainsi qu'à mes parents, toute ma famille et celle de Laura, pour m'avoir supporté et encouragé tout au long de cette étude. Ce mémoire leur est dédié.



# Table of Contents

<b>INTRODUCTION GENERALE . . . . .</b>	<b>1</b>
<b>CHAPTER 1 : MODEL-BASED DIAGNOSIS OF NONLINEAR FLAT SYSTEMS . . . . .</b>	<b>5</b>
1.1 INTRODUCTION . . . . .	7
1.2 DIFFERENTIAL FLATNESS OF NONLINEAR SYSTEMS . . . . .	8
1.2.1 Preliminary notions . . . . .	8
1.2.1.1 Flatness necessary and sufficient conditions . . . . .	10
1.2.2 Flat outputs processing of a permanent magnet stepper motor . . . . .	15
1.2.2.1 Types of stepper motors . . . . .	17
1.2.2.2 Stator based (a,b)-model . . . . .	19
1.2.2.3 Rotor-based (d,q)-model . . . . .	23
1.2.2.4 Flat outputs processing . . . . .	25
1.2.3 Linearisation of flat systems . . . . .	28
1.2.3.1 Coordinate change and static feedback linearisation . . . . .	29
1.2.3.2 Endogenous dynamic feedback linearisation . . . . .	33
1.2.3.3 Pseudo-linearisation . . . . .	34
1.2.3.4 Static feedback linearisation of a hybrid stepper motor . . . . .	38
1.3 MODEL-BASED MONITORING . . . . .	40
1.3.1 Residual generation . . . . .	44
1.3.1.1 The Fundamental Problem of Residual Generation (FPRG) . . . . .	44
1.3.1.2 Fault modelling . . . . .	45
1.3.2 Fault detection . . . . .	46
1.3.3 Fault isolation . . . . .	48
1.3.3.1 Parity space . . . . .	50
1.3.4 Linear observers . . . . .	51
1.3.4.1 Observability . . . . .	51
1.3.4.2 The Luenberger Observer . . . . .	54
1.3.4.3 The Standard Kalman Filter (SKF) . . . . .	56
1.3.4.4 Unknown Input Observers (UIO) . . . . .	57
1.3.5 Observers for nonlinear control-affine systems . . . . .	58
1.3.5.1 High Gain Observers . . . . .	58
1.3.5.2 Sliding-mode observers . . . . .	59
1.3.5.3 The Extended Kalman Filter (EKF) . . . . .	60
1.3.6 Robustness and performances evaluation . . . . .	61
1.4 CONCLUSION . . . . .	62
<b>CHAPTER 2 : PROPOSAL OF A FLIGHT-CRITICAL MONITORING SYSTEM . . . . .</b>	<b>63</b>
2.1 INTRODUCTION . . . . .	65
2.2 FLIGHT CRITICAL SYSTEMS OVERVIEW . . . . .	66
2.2.1 Definitions . . . . .	66
2.2.2 Aviation safety standards and airworthiness . . . . .	72
2.2.2.1 Regulation authorities . . . . .	72
2.2.2.2 Sources of specifications and recommended practices . . . . .	72



2.2.3	System engineering methodologies . . . . .	74
2.2.3.1	Leading System Engineering Methodologies . . . . .	76
2.2.4	Leading system engineering tools . . . . .	77
2.3	SAFETY ASSESSMENT OF A TURBOSHAFT FUEL VALVE ACTUATOR . . . . .	80
2.3.1	Current system description . . . . .	80
2.3.2	Failure Modes and Effects Analysis . . . . .	82
2.3.3	Fault Tree Analysis (FTA) . . . . .	85
2.3.4	Failure rate processing . . . . .	88
2.3.5	Development Assurance Level assessment . . . . .	88
2.3.6	Current actuator redundancies in flight control . . . . .	90
2.3.6.1	Flight control surfaces . . . . .	90
2.3.6.2	Flight control actuator redundancies . . . . .	91
2.3.6.3	Triple Modular Redundancy: the Triplex-AND voter . . . . .	92
2.3.6.4	Quadriplex-dual redundant actuator architecture . . . . .	93
2.4	PROPOSAL OF A SAFE AND ROBUST ARCHITECTURE . . . . .	94
2.5	CONCLUSION . . . . .	97
<b>CHAPTER 3 : EXPERIMENTAL SETUP . . . . .</b>		<b>99</b>
3.1	INTRODUCTION . . . . .	101
3.2	ACTUATOR MODEL VALIDATION AND INTEGRATION ON TEST BENCH . . . . .	102
3.2.1	Test bench design . . . . .	102
3.2.2	Path planning of control inputs by dynamic inversion . . . . .	103
3.2.3	Dynamic inversion based model linearisation . . . . .	106
3.2.4	PWM signals processing for power board inputs . . . . .	108
3.2.5	Sensors measurements validation . . . . .	109
3.2.5.1	<i>Sensor noise filtering</i> . . . . .	110
3.2.5.2	<i>Initial position settings</i> . . . . .	111
3.2.6	Stator fault generation . . . . .	113
3.2.6.1	<i>Realisation of a shorted stator winding</i> . . . . .	113
3.2.6.2	<i>Faults scenarios</i> . . . . .	113
3.3	DIAGNOSIS ALGORITHMS INTEGRATION AND VALIDATION ON BENCH . . . . .	116
3.3.1	Residuals generation based on dynamic inversion and a standard Kalman filter . . . . .	116
3.3.1.1	<i>Validation of linear system inputs reconstruction</i> . . . . .	117
3.3.1.2	<i>Measurement estimates validation</i> . . . . .	119
3.3.2	Diagnosis models comparison . . . . .	121
3.3.2.1	<i>Residual thresholds crossing</i> . . . . .	121
3.3.3	Conclusion . . . . .	124
<b>CONCLUSION AND PERSPECTIVES . . . . .</b>		<b>127</b>
<b>APPENDICES . . . . .</b>		<b>130</b>
APPENDIX A: DIFFERENTIAL GEOMETRY NOTIONS . . . . .		130
A.1.	Differentiable manifold . . . . .	131
A.2.	Diffeomorphism . . . . .	131
A.3.	Implicit Function Theorem . . . . .	131
A.4.	Module on a commutative ring . . . . .	132
A.5.	Particular matrices . . . . .	132
A.6.	Trivial Cartan field . . . . .	133
A.7.	Tangent space and tangent bundle . . . . .	133
A.8.	Differential form . . . . .	135

APPENDIX B: STEPPER MOTOR DATASHEETS .....	135
B.1. Model parameters .....	137
APPENDIX C: TEST BENCH SPECIFICATIONS .....	138
C.1. Hardware inputs and outputs .....	139
C.2. Bench wiring diagram .....	140
C.3. Power board specifications .....	141
C.4. Real time machine specifications .....	141
C.5. Controllable switches: Reed relays .....	143
<b>REFERENCES .....</b>	<b>155</b>

# Introduction générale

Cette thèse s'inscrit dans la thématique de recherche du groupe Automatique de l'IMS portant sur le diagnostic et la commande des systèmes pour l'aéronautique et le spatial. Cette étude vise à concevoir et développer des architectures et des algorithmes pour la commande et le diagnostic des systèmes aéronautiques critiques. L'innovation porte sur une méthodologie de conception d'algorithmes de commande tolérante aux défauts et de diagnostic à base de modèle non linéaire prenant en compte les contraintes du processus de certification aéronautique. Au niveau applicatif, une rupture technologique a été réalisée pour systèmes critiques de vol tel que les calculateurs embarqués. Le projet SYRENA (Safran-Turboméca) a constitué le vecteur d'application et accentué la synergie entre le secteur de la recherche et du développement industriel ainsi que les laboratoires de recherche académique.

## **Rappel de la problématique industrielle:**

Les produits THALES concernés par l'application de ces travaux de recherche sont essentiellement constitués par les calculateurs des systèmes de régulation moteur (principalement turbines d'hélicoptères), les EECU (Electronic Engine Control Unit) appelés parfois FADEC (Full Authority Digital Engine Control) et les actionneurs du circuit air-carburant. Ces produits ont en commun d'être à proximité du dispositif à piloter donc dans des environnements fortement contraints (EECU et actionneurs situés sur le bloc moteur) et font parties des chaînes critiques du système avionique. Un dysfonctionnement de l'un de ces produits ne doit pas conduire à un événement catastrophique et doivent donc être tolérants aux défauts. Ces contraintes conduisent aujourd'hui à des architectures complexes incluant des dissimilarités de conception et comprenant des redondances matérielles. A ces contraintes s'ajoute une disparité des actionneurs à piloter (vérins hydrauliques, électriques, électrovanne) et des

chaines de commande et d'acquisition principalement analogiques plus sensibles aux perturbations électromagnétiques. Cela conduit d'une part à un circuit de câblage important entre les capteurs/actionneurs et le calculateur, et à une chaîne de conditionnement et de traitement du signal plus complexe. La simplification des interfaces entre capteurs/actionneurs et le calculateur doit permettre de réduire la complexité des calculateurs (pour une éventuelle intégration dans le système avionique), de diminuer la masse du harnais et d'améliorer la fiabilité globale du système (sous respect des exigences de safety). De ce point de vue, l'intégration des fonctions d'asservissement et de surveillance au sein même de l'actionneur et l'utilisation de bus numérique constituent une piste à explorer. Ces technologies bien que déjà présentes sur le marché industriel sont très peu appliquées au secteur aéronautique en raison de la complexité du processus de certification.

### **Contenu du premier chapitre: diagnostic à base de modèles des systèmes différentiels non-linéaires plats**

Le chapitre 1 présente dans une première partie un rappel des propriétés des systèmes non-linéaires plats. Une application sur un modèle non-linéaire d'un moteur pas-à-pas a été réalisée. Une linéarisation par difféomorphisme et bouclage endogène est appliquée au système proposé dans le but d'utiliser son modèle linéaire équivalent au sein d'algorithmes de diagnostic à base d'observateurs tel que le filtre de Kalman linéaire.

Dans la deuxième sous-partie, un état de l'art des méthodes de diagnostic de défauts est présenté. Les générateurs de résidus tels que l'espace de parité et les observateurs présentent un intérêt particulier pour des applications embarquées au sein de systèmes critiques de vol tels que les calculateurs de commande moteur.

## Contenu du deuxième chapitre: proposition d'une architecture de surveillance d'un système critique de vol

Dans le chapitre 2, une architecture de surveillance de système critique de vol est proposée et à fait l'objet d'un brevet. Un rappel sur la problématique de la criticité et de la sûreté de fonctionnement aéronautique est proposé en première partie. Afin de prendre en compte la contrainte de sûreté de fonctionnement dans la conception d'un système aéroporté, les méthodes, outils et normes imposés par les organismes d'aéronavigabilité, tels que la FAA, l'EASA et l'ICAO, sont présentés. Un exemple d'allocation du niveau de sûreté d'un actionneur de commande moteur d'hélicoptère est réalisé. L'innovation de cette étude porte sur la proposition d'une architecture calculateur permettant de réduire l'encombrement lié aux redondances matérielles en remplaçant ces composants par des fonctions analytiques embarquées sur calculateur. La dualité: redondance matérielle et analytique est étudiée.

## Contenu du troisième chapitre: réalisation du procédé expérimental

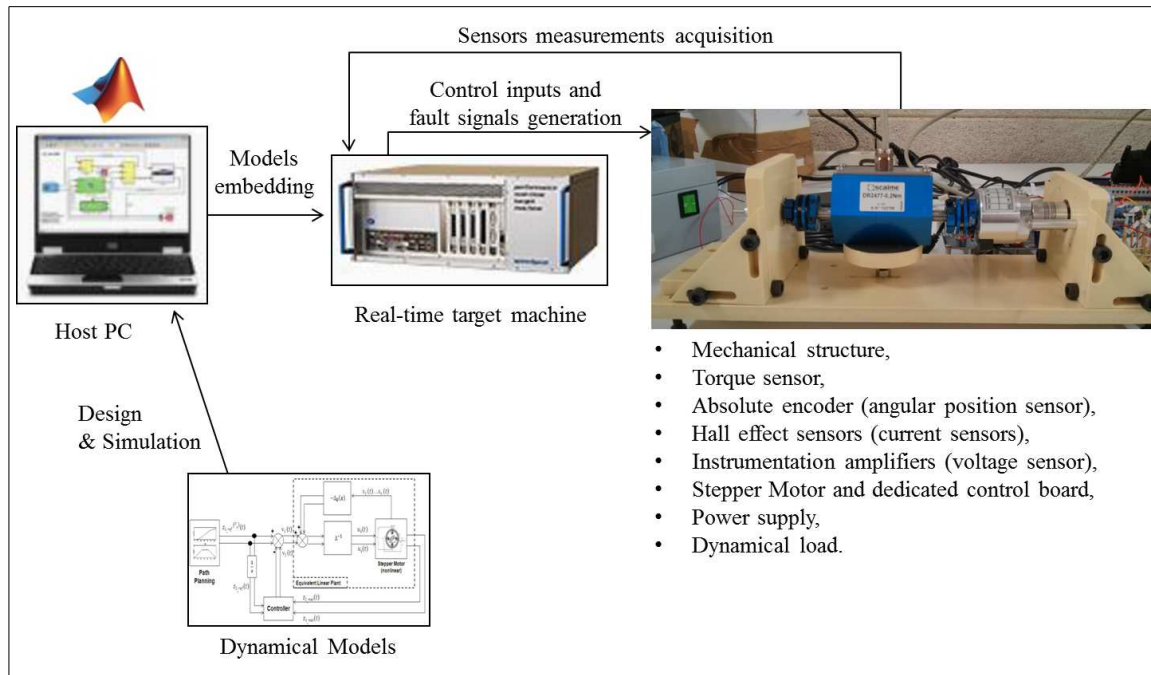


Figure 1: Schéma de principe du démonstrateur technique réalisé

Le chapitre 3 est dédié à l'élaboration d'un démonstrateur technique dont le but est de détecter les défauts courts-circuit d'une phase statorique d'un moteur pas-à-pas hybride. Le moteur électrique similaire à celui de l'actionneur de vanne de dosage du circuit carburant de turbine d'hélicoptère, a été câblé de façon à pouvoir générer des courts-circuits sur différents pourcentages de spires statoriques. Les algorithmes développés dans le chapitre 1 ont été réalisés en simulation puis embarqués sur une machine temps réel elle-même connectée aux capteurs du banc d'essai (figure 1). Dans la dernière partie du chapitre 3, les performances de deux méthodes de diagnostic à base d'observateurs de Kalman sont comparées. La première fait appel à une linéarisation par difféomorphisme et bouclage endogène. Les résidus sont obtenus avec un filtre de Kalman linéaire. La seconde méthode fait appel à un filtre de Kalman étendu (EKF) dont le procédé de linéarisation n'est pas exact générant des erreurs de linéarisation.

# Chapter 1

## Model-based diagnosis of nonlinear flat systems

### Contents

---

<b>1.1</b>	<b>Introduction . . . . .</b>	<b>7</b>
<b>1.2</b>	<b>Differential flatness of nonlinear systems . . . . .</b>	<b>8</b>
1.2.1	Preliminary notions . . . . .	8
1.2.2	Flat outputs processing of a permanent magnet stepper motor . . . . .	15
1.2.3	Linearisation of flat systems . . . . .	28
<b>1.3</b>	<b>Model-based monitoring . . . . .</b>	<b>40</b>
1.3.1	Residual generation . . . . .	44
1.3.2	Fault detection . . . . .	46
1.3.3	Fault isolation . . . . .	48
1.3.4	Linear observers . . . . .	51
1.3.5	Observers for nonlinear control-affine systems . . . . .	58
1.3.6	Robustness and performances evaluation . . . . .	61
<b>1.4</b>	<b>Conclusion . . . . .</b>	<b>62</b>

---

---

**Abstract:** In this chapter, it is shown that differential flat systems present useful properties for model-based fault diagnosis methods. At first, the principles of nonlinear and multivariable flat systems are recalled in section 1.2. Dynamical inversion properties are considered for the linearisation of flat systems, such as permanent magnet stepper motors (PMSM). Next, the state of the art in current analytical diagnosis methods is described in section 1.3, particularly addressing residual analysis methods. Observers and parity space methods are then detailed for linear and nonlinear systems.

---



## 1.1 Introduction

The occurrence of faults on physical systems may downgrade their performance or in the worst case lead to catastrophic events. Current control systems are designed to ensure their stability and robustness but in the case of high-integration complex aircraft systems e.g., engines, flight-control actuators or flight computer units (FCU), monitoring functions are also required. Among current monitoring solutions for engine and flight control computer units, material and analytic redundancy allow faults detection and isolation (FDI) to prevent from catastrophic events but also to optimise maintenance. Material redundancy impacts the number of sensors and physical components on board of an aircraft and require more space, weight and power (SWaP). Therefore the use of analytical models in monitoring systems has increased and may replace rows of redundant components in future aircraft systems. This chapter is devoted to the presentation of model-based techniques for FDI purposes [26], where models are considered as a set of differential equations describing a physical system. Regarding FDI, fault-detection is based on the comparison of sensor measurements and the measurement estimation processed by the mathematical model.

In section (1.2), a special attention is dedicated to the properties of nonlinear and multidimensional flat systems. Two groups of nonlinear systems exist, namely:

- Strong nonlinear systems, for which specific analysis and processing tools are necessary,
- Pseudo-nonlinear systems [45], which are equivalent to linear systems by coordinate change and feedback,

but in our case, the study will deal only with pseudo-nonlinear systems. A nonlinear system described by differential equations is differentially flat [41] if its dynamical behaviour can be expressed by a set of nonlinear and smooth functions which are also differentially independent. These functions are the "flat outputs" of the system.

They depend on the states of the system and a finite number of the inputs time derivatives. Therefore, each trajectory of a nonlinear system within the state space can be determined by knowing its flat outputs without processing integral functions on the dynamic of the system. The main objective of differential flatness is to determine a set of flat outputs which permits a nonlinear dynamic inversion (NLDI) [59] without generating a non observable zeros dynamic of the system.

In order to be compliant with linear diagnosis models, linearisation properties of flat systems based on coordinate change and endogenous feedback are used to determine the linear equivalent system of a permanent magnet stepper motor (PMSM) model. In section (1.3), model-based monitoring models are recalled. Residual generation methods are shown, including parity space, linear and nonlinear observers such as the Standard Kalman Filter (SKF) and the Extended Kalman Filter(EKF). Robustness and performances of NLDI and linearisation are discussed regarding the tangent linearisation of the EKF.

## **1.2 Differential flatness of nonlinear systems**

The concept of differential flat systems was introduced in the middle of the 1990's where the first application studies were realised in [20] for aerospace projects. The development of the theory continued within the work of P.Martin [80] who contributed to the formal concept of flatness presented by M. Fliess et al in [41]. Since then, this concept has known many applications such as: robust system control [63, 55, 49, 77, 78], fault tolerant control [58, 107], path planning [101, 85, 33, 104], fault detection and diagnosis (FDD & FDI) [81, 39] and estimation of nonlinear parameters [40].

### **1.2.1 Preliminary notions**

To begin with, usual definitions of flat systems initially introduced by Fliess et al. in [41] for nonlinear systems ruled by ordinary differential equations (ODE) are

recalled. A nonlinear system is flat if there exist a set of differentially independent variables called flat outputs. Their number equals to the number of inputs where the states and the inputs can be expressed with the flat outputs and a finite number of their time derivatives. As a consequence, control inputs and states can be processed by planning only the trajectories of the flat outputs. Trajectory planning using flatness properties was applied in [85, 104] and in the design of robust controllers [55, 49].

**Definition 1.1. (Differential flatness)**

*Considering the following nonlinear multivariable system:*

$$\dot{x} = f_n(x_n, u_n) \quad (1.1)$$

*with*

$$\begin{aligned} x_n &= (x_{n_1}, \dots, x_{n_n}), x_n \in \mathbb{R}^n \\ u_n &= (u_{n_1}, \dots, u_{n_m}), u_n \in \mathbb{R}^m \end{aligned} \quad (1.2)$$

*and*

$$f_n = (f_{n_1}, \dots, f_{n_m}) \quad (1.3)$$

*is a smooth function of  $x_n$  and  $u_n$  satisfying*

$$f_n(0, 0) = 0 \quad (1.4)$$

*where*

$$\text{rank} \left\{ \frac{df_n}{du_n}(0, 0) \right\} = m \quad (1.5)$$

*System (1.1) is differentially flat if there exists an output vector  $z_n$  named flat output, composed of  $m$  fictive outputs such as:*

- *State vector  $x_n$  and control input vector  $u_n$  can be expressed with the flat outputs:*

$$z_n = (z_1, \dots, z_m) \quad (1.6)$$

and a finite number of their derivatives, as shown in equations (1.7).

- Flat outputs  $z$  are expressed with the state vector  $x$ , the control vector  $u$  and a finite number of their time derivatives. These two conditions are given by:

$$\begin{aligned} x &= B(z, \dot{z}, \dots, z^{(q)}) \\ u &= C(z, \dot{z}, \dots, z^{(r)}) \\ z &= A(x, u, \dot{u}, \dots, u^{(p)}) \end{aligned} \tag{1.7}$$

where  $p$ ,  $q$  and  $r$  are integers,  $z^{(q)}$  is the  $q^{th}$  time derivative of  $z$  and

$A = (A_1, \dots, A_n), B = (B_1, \dots, B_m), C = (C_1, \dots, C_m)$  are smooth mappings.

**Remark 1.1.** Given a flat system, the number of components of a flat output is equal to the number of independent inputs.

### 1.2.1.1 Flatness necessary and sufficient conditions

In this section, flat outputs processing algorithms introduced in [75] are applied. Another processing method based on modules was described in [89]. The following flatness necessary and sufficient conditions were established by J.Lévine in [73] regarding nonlinear systems.

In order to process flat outputs, external and local flatness properties were established. The external flatness formulation definitions is based on the Lie-Bäcklund equivalence of two implicit systems.

Assuming  $X$  a differentiable manifold of dimension  $n$ ,  $T_x X$  the corresponding tangent space at  $x \in X$  and  $TX = \bigcup_{x \in X} T_x X$  its tangent bundle (see Appendix A.7).

The nonlinear implicit system<sup>1</sup> (see Appendix A.3) is given by:

$$F(x, \dot{x}) = 0 \tag{1.8}$$

---

<sup>1</sup>The inputs  $u$  do not appear in the implicit system.

where  $F$  is a  $C^\infty$  mapping from  $TX$  to  $\mathbb{R}^{n-m}$  in a given neighbourhood of  $TX$  with  $\text{rank} \left( \frac{\partial F}{\partial \dot{x}} \right) = n - m$ . Also,

$$\mathfrak{X} \triangleq X \times \mathbb{R}_\infty^n = X \times \mathbb{R}^n \times \mathbb{R}^n \times \dots \quad (1.9)$$

is considered as the manifold of infinite order jets [42], defined as the product of  $X$  with an infinite amount of  $\mathbb{R}^n$  and its general coordinates  $\bar{x}$  such as:

$$\bar{x} = (x_1, \dots, x_n, \dot{x}_1, \dots, \dot{x}_n, \dots, x_1^{(k)}, \dots, x_n^{(k)}, \dots). \quad (1.10)$$

The jets of infinite order allow expression of system (1.8) with its implicit equations given in definition 1.2.

**Definition 1.2. (Implicit system)** *An implicit system is composed of a triplet  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  with:*

$$\mathfrak{X} = X \times \mathbb{R}_\infty^n \quad (1.11)$$

and

$$\tau_{\mathfrak{X}} = \sum_{i \geq 0} \sum_{j=1}^n x_j^{(i+1)} \frac{\partial}{\partial x_j^{(i)}} \quad (1.12)$$

its associated trivial Cartan field (see Appendix A.6), and  $F \in C^\infty(TX; \mathbb{R}^{n-m})$ , satisfying:

$$\text{rank} \left( \frac{\partial F}{\partial \dot{x}} \right) = n - m \quad (1.13)$$

for a given neighbourhood of  $TX$ .

Considering two implicit systems  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  and  $(\mathfrak{D}, \tau_{\mathfrak{D}}, G)$  with  $\mathfrak{D} = Y \times \mathbb{R}_\infty^p$ ,  $\dim X = n$ ,  $\dim Y = p$  and  $\text{rank} \left( \frac{\partial G}{\partial \dot{y}} \right) = p - q$ , their equivalence conditions is given by the Lie-Bäcklund equivalence definition as follows.

**Definition 1.3. (Lie-Bäcklund equivalence)** *The two implicit controlled systems  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  and  $(\mathfrak{D}, \tau_{\mathfrak{D}}, G)$  are Lie-Bäcklund equivalent (or L-B equivalent) at  $(\bar{x}_0, \bar{y}_0) \in \mathfrak{X}_0 \times \mathfrak{D}_0$  if and only if:*

- there exist neighbourhoods  $\mathcal{X}_0$  and  $\mathcal{Y}_0$  of  $\bar{x}_0 \in \mathfrak{X}_0$  and  $\bar{y}_0 \in \mathfrak{D}_0$  respectively and a mapping

$$\Phi = (\psi_0, \psi_1, \dots) \in C^\infty(\mathcal{X}_0; \mathcal{Y}_0) \quad (1.14)$$

such as  $\Phi(\bar{y}_0) = \bar{x}_0$  and<sup>2</sup>  $\Phi_*\tau_{\mathfrak{D}} = \tau_{\mathfrak{X}}$ ,

- there exist a mapping

$$\Psi = (\psi_0, \psi_1, \dots) \in C^\infty(\mathcal{X}_0; \mathcal{Y}_0) \quad (1.15)$$

such as  $\Psi(\bar{x}_0) = \bar{y}_0$  and  $\Psi_*\tau_{\mathfrak{X}} = \tau_{\mathfrak{D}}$

$\Psi$  and  $\Phi$  are the Lie-Bäcklund isomorphisms and invertible at  $(\bar{x}_0, \bar{y}_0)$ . Both controlled systems  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  and  $(\mathfrak{D}, \tau_{\mathfrak{D}}, G)$  are locally L-B equivalent if:

$$(\bar{x}, \Psi(\bar{x})) = (\Phi(\bar{y}), \bar{y}) \quad (1.16)$$

The system  $(X \times U \times \mathbb{R}_\infty^m, \bar{f})$ , resp.  $(X \times \mathbb{R}_\infty^n, \tau_X, F)$ , where  $\bar{f} = (f, \dot{f}, \ddot{f})$  with  $m$  inputs, is differently flat if and only if it is L-B equivalent to the trivial system  $(\mathbb{R}_\infty^m, \tau_m)$ , resp.  $(\mathbb{R}_\infty^m, \tau_m, 0)$ , where  $\tau_m$  is the Cartan field of  $\mathbb{R}_\infty^m$  such as:

$$\tau_m = \sum_{j \geq 0} \sum_{i=1}^m y^{(j+1)} \frac{\partial}{\partial y_i^{(j)}} \quad (1.17)$$

$y$  is called the flat output vector.

The flatness of the implicit system  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  corresponds to the fact that it is Lie-Bäcklund equivalent to the trivial system  $(\mathbb{R}_\infty^m, \tau_m, 0)$ .

**Definition 1.4. (Flatness, External Formulation [73])** *The implicit system  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  is flat at  $(\bar{x}_0, \bar{y}_0) \in \mathfrak{X}_0 \times \mathbb{R}_\infty^m$  if and only if it is L-B equivalent at  $(\bar{x}_0, \bar{y}_0) \in \mathfrak{X}_0 \times \mathbb{R}_\infty^m$  to the trivial implicit system  $(\mathbb{R}_\infty^m, \tau_m, 0)$ . In this case, the Lie-Bäcklund isomorphisms  $\Phi$  and  $\Psi$  are called uniformization, as referred to Hilbert's 22<sup>nd</sup> Problem [8]*

---

<sup>2</sup> $\Phi_*\tau_{\mathfrak{D}}$  is the image of the Cartan field  $\tau_{\mathfrak{D}}$  by the mapping  $\Phi$

The extension of the flatness external formulation to local flatness is given by the following definition.

**Theorem 1.1. (Local flatness)** *The implicit system  $(\mathfrak{X}, \tau_{\mathfrak{X}}, F)$  is flat at  $(\bar{x}_0, \bar{y}_0)$  with  $\bar{x}_0 \in \mathfrak{X}_0$  and  $\bar{y}_0 \in \mathbb{R}_{\infty}^m$  if and only if a local invertible smooth mapping  $\Phi$  of  $\mathbb{R}_{\infty}^m$  to  $\mathfrak{X}_0$  exists, with a smooth inverse such as:*

$$\Phi(\bar{y}_0) = \bar{x}_0 \quad (1.18)$$

and

$$\Phi^* dF = 0 \quad , \quad (1.19)$$

where  $\Phi^*$  denotes the set of invertible elements of  $\Phi$ .

Considering the definition of the following polynomial matrices:

$$P(F) = \frac{\partial F}{\partial x} + \frac{\partial F}{\partial \dot{x}} \frac{d}{dt} \quad \text{and} \quad P(\varphi_0) = \sum_{j \geq 0} \frac{\partial \varphi_0}{\partial y^{(j)}} \frac{d^j}{dt^j} \quad , \quad (1.20)$$

equation (1.19) becomes:

$$\Phi^* dF = P(F)P(\varphi_0)dy = 0. \quad (1.21)$$

Linear algebraic methods for polynomial matrices such as the following Smith decomposition, allows description of a variational system  $P(F)$  given by:

$$VP(F)U = (I_{n-m}, 0_{n-m,m}). \quad (1.22)$$

where  $n$  is the rank of the system and  $m$  the dimension of the output vector.

Assuming  $\mathfrak{K}$ , the ring of meromorphic<sup>3</sup> functions from  $\mathfrak{X}$  to  $\mathbb{R}$ ,  $\mathfrak{K} \left[ \frac{d}{dt} \right]$  the ring of polynomials in  $\frac{d}{dt}$  with coefficients in  $\mathfrak{K}$ , and  $\mathfrak{M}_{n,m} \left[ \frac{d}{dt} \right]$  the module (see Appendix

---

<sup>3</sup>A meromorphic function on an open subset  $D$  of the complex plane is a function that is infinitely differentiable and equal to its own Taylor series on all  $D$  except a set of isolated points, which are poles for the function.

A.4) of the matrices of dimension  $n \times m$  on  $\mathfrak{K} \left[ \frac{d}{dt} \right]$ . Matrices of  $\mathfrak{M}_{n,n} \left[ \frac{d}{dt} \right]$  possessing an inverse in  $\mathfrak{M}_{n,n} \left[ \frac{d}{dt} \right]$  are called unimodular matrices (see Appendix A.5) and form a subgroup  $\mathcal{U}_n \left[ \frac{d}{dt} \right]$  of  $\mathfrak{M}_{n,n} \left[ \frac{d}{dt} \right]$ . Using the previous notations, the matrices  $P(\varphi_0) \in \mathfrak{M}_{n-m,n} \left[ \frac{d}{dt} \right]$  satisfying equation (1.21) can be fully determined by solving the following matrix equation:

$$P(F)\Theta dy = 0, \quad (1.23)$$

where  $\Theta$  is a hyper-regular matrix (see Appendix A.5).

**Lemma 1.1.** *Hyper-regular matrices  $\Theta \in \mathfrak{M}_{n,m} \left( \frac{d}{dt} \right)$  satisfying (1.23) are given by:*

$$\Theta = U \begin{pmatrix} 0_{n-m,m} \\ I_m \end{pmatrix} W, \quad (1.24)$$

where  $U \in R\text{-Smith}(P(F))$  and  $W \in \mathcal{U}_m \left( \frac{d}{dt} \right)$ .

Assuming:

$$\hat{U} = U \begin{pmatrix} 0_{n-m,m} \\ I_m \end{pmatrix}. \quad (1.25)$$

**Lemma 1.2.** *For a given matrix  $Q$  such as  $Q \in L\text{-Smith}(\hat{U})$ , there exists a matrix  $Z \in \mathcal{U}_m \left( \frac{d}{dt} \right)$  such as:*

$$Q\Theta = \begin{pmatrix} I_m \\ 0_{n-m,m} \end{pmatrix} Z. \quad (1.26)$$

A necessary and sufficient condition for the implicit system (1.8) to be flat at  $\bar{x}_0 \in \mathfrak{X}_0$  is that the  $\mathfrak{K} \left[ \frac{d}{dt} \right]$ -ideal, generated by the 1-forms  $\omega, \dots, \omega_m$  (see Appendix A.8) defined by:

$$\omega(\bar{x}) = \begin{pmatrix} \omega_1(\bar{x}) \\ \vdots \\ \omega_m(\bar{x}) \end{pmatrix} = (I_m, 0_{m,n-m}) Q(\bar{x}) dx|_{x_0} = \tilde{Q}(\bar{x}) dx|_{x_0}. \quad (1.27)$$



is strongly closed in a neighbourhood of  $\bar{x}_0 \in \mathfrak{X}_0$ . A flat output of the variational system is obtained if  $d\omega = 0$  by integration of  $dy$  where:

$$dy = \omega. \quad (1.28)$$

**Definition 1.5. (Strongly closed set)** If  $\tau_1, \dots, \tau_r$  are  $r$  independent 1-forms, the  $\mathfrak{K} \left[ \frac{d}{dt} \right]$ -ideal generated by  $\tau_1, \dots, \tau_r$  is the set made of the combinations with coefficients in  $\mathfrak{K} \left[ \frac{d}{dt} \right]$  of the forms  $\eta \wedge \tau_i$  with  $\eta$  arbitrary form of an arbitrary degree on  $\mathfrak{X}_0$  and  $i = 1, \dots, r$ . This ideal is said to be strongly closed if and only if there exists  $M \in \mathcal{U}_r \left[ \frac{d}{dt} \right]$  such that  $d(M\tau) = 0$  with  $\tau = (\tau_1, \dots, \tau_r)^T$ .

If  $d\omega \neq 0$ , then it is necessary to find a base for which (1.28) can be integrated. Therefore an integral factor  $M \in \mathcal{U}_m(\frac{d}{dt})$  verifying  $d(M\omega) = 0$  has to be found. The  $\mathfrak{K} \left[ \frac{d}{dt} \right]$ -ideal  $\Omega$ , generated by the 1-forms (defined by (1.27)) is strongly closed in  $\chi_0$  if and only if an operator  $\mu \in \mathcal{L}_1((\Lambda(\mathfrak{X}))^m)$ , and a matrix  $M \in \mathcal{U}_m(\frac{d}{dt})$  exist such that:

$$d\omega = \mu\omega, \quad \mathfrak{d}(\mu) = \mu^2, \quad \mathfrak{d}(M) = -M\mu \quad (1.29)$$

where  $\mathcal{L}_1((\Lambda(\mathfrak{X}))^m)$  is the space of linear operators mapping the  $p$ -forms of dimension  $m$  of  $\mathfrak{X}$ , in  $(p+1)$ -forms of dimension  $m$  in  $\mathfrak{X}$ .  $\mathfrak{d}$  is the extension of the exterior derivative  $d$ , with coefficients in  $\mathfrak{K} \left[ \frac{d}{dt} \right]$ .

As an example, the presented flat outputs processing method based on Smith decomposition and 1-forms calculation is applied in the next section to a nonlinear permanent magnet stepper motor (PMSM) model.

### 1.2.2 Flat outputs processing of a permanent magnet stepper motor

Before processing the flat outputs of the PMSM, some fundamentals regarding the different types of stepper motors are recalled. Stepper motors are electromechanical converters which aim is to transform an electrical power into a linear or angular

motion (Figure 1.1).

From the electro-technical point of view, its structure is very similar to the syn-

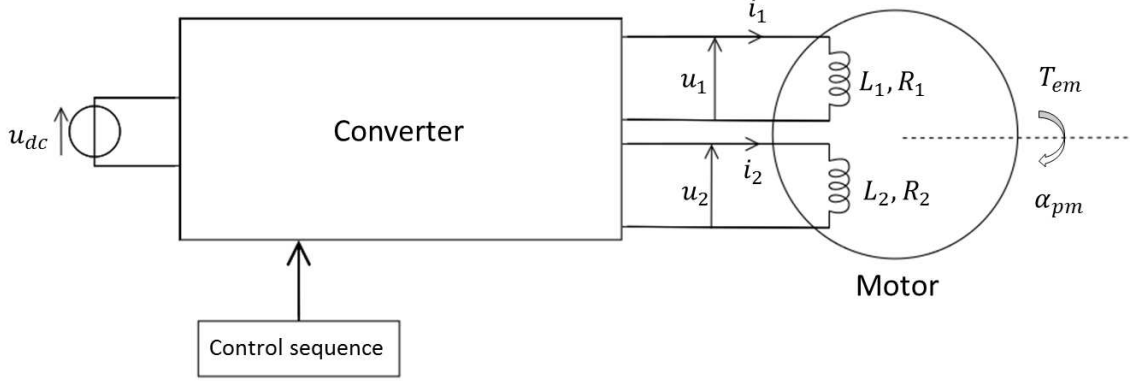


Figure 1.1: Electromechanical conversion scheme of a two-phased stepper motor [86]

chronous motor. Windings encircle the stator poles (mostly made of salient poles) and the rotor can be made of permanent magnets (also called polarised or active structure), or ferromagnetic parts (also called reluctant or passive structure). Three main types of stepper motors exist:

- the variable reluctance stepper motor (VRSM), as shown in figure 1.2,
- the permanent magnet stepper motor (PMSM), as shown in figure 1.3,
- the hybrid stepper motor (HSM), as shown in figure 1.4.

The power supply of each winding generates a current  $i$  which generates a magnetic field in a specific direction. Powering up each winding in a given sequence allows to move the magnetic field of the stator along an elementary resolution called step. The variation of the power sequence on each stator winding defines a rotating magnetic field corresponding to a full step, half step or microstep. The discrete motion of the magnetic field allows the rotor to rotate with a synchronising torque. For each type of stepper motor, the torque is generated:

- by the stator field (current)-rotor iron interaction, in the case of a passive rotor (Figure 1.2),

- by the stator field (current)-rotor (magnetic) field interaction, in the case of a permanent magnet rotor (Figure 1.3),
- by the two previous magnetic field interactions at the same time in the case of a hybrid stepper motor (Figure 1.4).

### 1.2.2.1 Types of stepper motors

In the case of a passive rotor (Figure 1.2), when the current goes through winding 1, the established magnetic field places the iron of the rotor in a position corresponding to a maximum magnetic flow. The pole of the rotor is then aligned with the stator

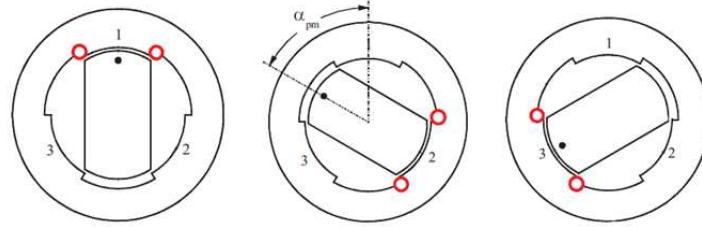


Figure 1.2: 3-phased variable reluctance stepper motor (VRSM), reference: support de cours, Haute École Spécialisée de Suisse Occidentale (HESSO), M. Correvon.

electromagnets. When the next phase commutation occurs, the rotor rotates of a mechanical angle (step angle)  $\alpha_{pm}$ . The resulting synchronising torque from the "current-iron" interaction is related to the variation of self inductance of the supplied winding.

The difference of the three stepper motor types is related to the generation of the electromechanical torque  $T_{em}$  which will next be described. The inductance  $L$  of each stator phase is related to the reluctance<sup>4</sup>  $\mathfrak{R}$  and the number of coils  $n_c$  by:

$$L = \frac{n_c^2}{\mathfrak{R}}. \quad (1.30)$$

In an unsaturated regime, the global magnetic flow is given by:

$$\phi = \phi_f + L_j i_k + M_{kj} i_k, \quad (1.31)$$

<sup>4</sup>The reluctance  $\mathfrak{R}$  is related to the length of the material in which the magnetic flow runs, the section of the material and the material permeability.

where  $\phi_f$  is the inductor flow related to the magnets,  $L_j i_k$  and  $M_{kj} i_k$  are respectively the flows related to self and mutual inductances  $L$  and  $M$  of phases  $j$  and  $k$ .

In the case of a 2-phased VRSM, the expression of the torque depends only on the inductance of the electromagnets [86]. The torque is pure reluctant in this case and is given by:

$$T_{em} = \frac{1}{2} i_1^2 \frac{dL_1(\alpha_{pm})}{d\alpha_{pm}} + \frac{1}{2} i_2^2 \frac{dL_2(\alpha_{pm})}{d\alpha_{pm}} \quad (1.32)$$

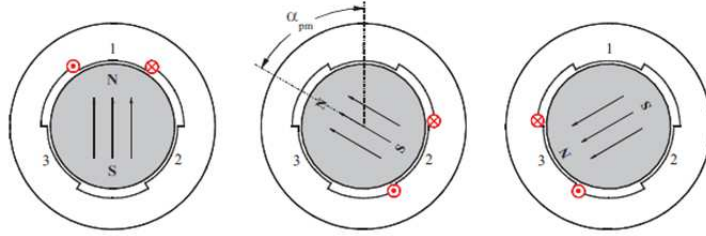


Figure 1.3: 3-phased permanent magnet stepper motor (PMSM), reference: support de cours, Haute École Spécialisée de Suisse Occidentale (HESSO), M. Correvon

where  $L_j$  and  $i_j$  are respectively the inductance and current of the stator phase  $j$ . The next considered magnetic flows  $\phi$  correspond to the flow generated by the cumulated coils of each stator phase.

When the rotor is made of a permanent magnets (Figure 1.3), each magnet has a constant permeability and the air gap is also constant ( $L$  and  $M$  are also constant). Therefore the synchronising electromagnetic torque  $T_{em}$  is only related to the flow  $\phi_{fj}$  variation of the  $j^{th}$  phase, between the rotor magnet and the active electromagnet of the stator. It is also called hybrid torque and its expression, for a two-phased motor is given by:

$$T_{em} = i_1 \frac{d\phi_{f1}}{dt} + i_2 \frac{d\phi_{f2}}{dt} , \quad (1.33)$$

The last polarised structure called hybrid (Figure 1.4), works mainly on the iron-iron interaction (variable reluctance) but with a permanent magnet (rotor) excitation. The torque of this type of motor contains three fundamental torques: the hybrid torque (main torque in this case) as shown in equation (1.33), the reluctant torque

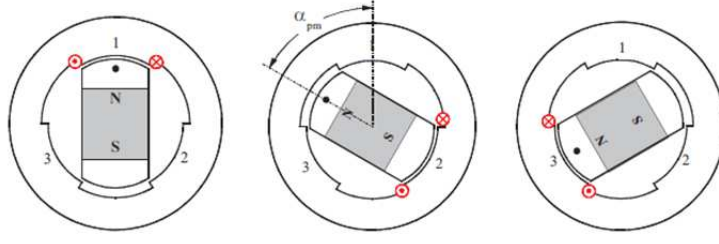


Figure 1.4: 3-phased hybrid stepper motor, reference: support de cours, Haute École Spécialisée de Suisse Occidentale (HESSO), M. Correvon.

given by equation (1.32) and the detent torque given by:

$$T_d = -K_d \sin(4n\alpha_{pm}) , \quad (1.34)$$

where  $n$  is the number of rotor teeth and  $K_d$  is the detent torque constant.

#### 1.2.2.2 Stator based (a,b)-model

For the following case, the inductances  $L_i$  are constant and the rotor speed  $\omega$  is constant. The nonlinear PMSM bi-phased model can be expressed with electrical and mechanical equations in the stator base  $(a,b)$  as follows [14]:

$$\begin{cases} u_a = Ri_a + n_c \frac{d\phi_a}{dt} \\ u_b = Ri_b + n_c \frac{d\phi_b}{dt} \end{cases} \quad (1.35)$$

where  $(u_a, u_b)$  and  $(i_a, i_b)$ , are respectively the voltages and currents. Subscripts  $a$  and  $b$  correspond respectively to  $A$  and  $B$  phases. The magnetic flows  $\phi_a$  and  $\phi_b$  depending on the cumulated coils of each phase are given by:

$$\begin{cases} \phi_a = \phi_{aa} + \phi_{ab} + \phi_{am} \\ \phi_b = \phi_{bb} + \phi_{ba} + \phi_{bm} \end{cases} \quad (1.36)$$

considering:

- $\phi_{aa}$  and  $\phi_{bb}$  the flow of each phase,
- $\phi_{ab}$  and  $\phi_{ba}$  the mutual flow between two phases,

- $\phi_{am}$  and  $\phi_{bm}$  the mutual magnet-electromagnet flows.

Because  $A$  and  $B$  phases are in a ninety degree displacement (phase quadrature), their mutual flows are equal to zero implying  $\phi_{ab} = 0$  and  $\phi_{ba} = 0$ . Therefore replacing the expression of the flow  $\phi$  of system (1.36) in system (1.35) allows to dissociate the voltage related to the main flow  $\phi$  onto two voltages such as for one phase:

$$n_c \frac{d\phi_a}{dt} = n_c \frac{d\phi_{aa}}{dt} + n_c \frac{d\phi_{am}}{dt} \quad (1.37)$$

where  $n_c \frac{d\phi_{aa}}{dt}$  is the voltage of the phase  $A$  winding and  $n_c \frac{d\phi_{am}}{dt}$  is the electromotive force (e.m.f)  $e_a$ . The inductance  $L$  is considered as a constant and is related to the flow  $\phi_{aa}$  and induction current  $i_a$  by the following equation:

$$L = n_c \frac{d\phi_{aa}}{di_a}. \quad (1.38)$$

Also, according to the Faraday law (in current generator mode),

$$u_a = n_c \frac{d\phi_{aa}}{dt}, \quad (1.39)$$

where  $u$  is the e.m.f generated by a flow passing through  $n_c$  coils. The development of equation (1.38) implies:

$$L = n_c \frac{d\phi_{aa}}{dt} \frac{dt}{di_a}. \quad (1.40)$$

Replacing equation (1.39) in equation (1.40) implies :

$$L \frac{di_a}{dt} = n_c \frac{d\phi_{aa}}{dt} \quad (1.41)$$

and therefore :

$$u_a = L \frac{di_a}{dt}. \quad (1.42)$$

Using equations 1.41 and 1.37 in the first equation of system 1.35 gives as a result:

$$u_a = Ri_a + L \frac{di_a}{dt} + n_c \frac{d\phi_{am}}{dt} \quad (1.43)$$

The last term of equation 1.43 corresponds to the e.m.f  $e_a$  of phase  $A$  given by:

$$e_a = n_c \frac{d\phi_{am}}{dt}. \quad (1.44)$$

According to the Boucherot law, in the case of sinusoidal input voltage  $u(t)$ , the flow  $\phi(t)$  has a sinusoidal wave form given by:

$$\phi(t) = \Phi_{max} \cos(n\theta_m) \quad (1.45)$$

Therefore, the e.m.f equations (1.44) is given by:

$$e_a = K_e \omega \sin(n\theta_m) , \quad (1.46)$$

where  $K_e$  is the e.m.f constant. Considering equations (1.46) and (1.41) and supposing the windings inductances constant, system (1.35) becomes:

$$\begin{cases} u_a = Ri_a + L \frac{di_a}{dt} - \omega \sin(n\theta_m) K_t \\ u_b = Ri_b + L \frac{di_b}{dt} + \omega \cos(n\theta_m) K_t \end{cases} \quad (1.47)$$

where  $K_t$  is the torque constant, considered equal to the e.m.f constant,  $n$  the number of rotor teeth,  $\omega$  the angular velocity of the rotor and  $\theta_m$  its angular position. The e.m.f  $e_a$  and  $e_b$  are given by:

$$\begin{cases} e_a = -\omega \sin(n\theta_m) K_t \\ e_b = \omega \cos(n\theta_m) K_t . \end{cases} \quad (1.48)$$

According to the fundamental principle of dynamics, the efforts are only related to rotating motions implying:

$$T_m - T_r = J \frac{d\omega}{dt} \quad (1.49)$$

with  $T_m$  the motorising torque and  $T_r$  the resisting torque, considering

$$T_m = T_{em} + T_d , \quad (1.50)$$

where  $T_d$  is the detent torque generated by the magnets. Without considering the mechanical losses, the electromechanical torque  $T_{em}$  is given by:

$$T_{em} = \frac{e_a i_a + e_b i_b}{\omega}. \quad (1.51)$$

The detent torque is equal to 10% of the global torque and is therefore neglected implying  $T_d = 0$ . According to equations (1.48) and (1.51), the electromechanical torque becomes:

$$T_{em} = [-K_t i_a \sin(n\theta_m) + K_t i_b \cos(n\theta_m)] \quad (1.52)$$

and the PMSM model satisfies:

$$\left\{ \begin{array}{lcl} u_a & = & R i_a + L \frac{di_a}{dt} - \omega \sin(n\theta_m) K_t \\ u_b & = & R i_b + L \frac{di_b}{dt} + \omega \cos(n\theta_m) K_t \\ J_m \frac{d\omega}{dt} & = & [-K_t i_a \sin(n\theta_m) + K_t i_b \cos(n\theta_m)] - B\omega \\ \frac{d\theta_m}{dt} & = & \omega \end{array} \right. \quad (1.53)$$

where  $B$  is the viscous friction coefficient and  $J_m$  the motor inertia.





$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = R \begin{bmatrix} i_a \\ i_b \end{bmatrix} + P(\theta_m) \begin{bmatrix} L & 0 \\ 0 & L \end{bmatrix} \left[ \frac{d\theta_m}{dt} \frac{d}{d\theta_m} (P(-\theta_m)) \begin{bmatrix} i_d \\ i_q \end{bmatrix} + P(-\theta_m) \frac{d}{d\theta_m} \begin{bmatrix} i_d \\ i_q \end{bmatrix} \right] + K_t \omega \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (1.57)$$

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = R \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} L & 0 \\ 0 & L \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + n\omega \begin{bmatrix} 0 & -L \\ L & 0 \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + K_t \omega \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (1.58)$$

The electromechanical torque  $T_m$  can also be expressed in the  $(dq)$ -base. Considering the park transformation matrix given by equation (1.54), the expression of  $i_q$  is:

$$i_q = -i_a \sin(n\theta_m) + i_b \cos(n\theta_m). \quad (1.59)$$

Replacing the expression of  $i_q$  in the electromechanical torque equation (1.52) implies:

$$T_{em} = K_t i_q \quad (1.60)$$

The motor model in the  $(dq)$  base is then given by:

$$\begin{cases} u_d &= R i_d + L \frac{di_d}{dt} - L n \omega i_q \\ u_q &= R i_q + L \frac{di_q}{dt} + L n \omega i_d + K_t \omega \\ J_m \frac{d\omega}{dt} &= K_t i_q - B \omega \\ \frac{d\theta_m}{dt} &= \omega \end{cases} \quad (1.61)$$

Considering the state vector:

$$x = \begin{pmatrix} i_d \\ i_q \\ \omega \\ \theta \end{pmatrix}, \quad (1.62)$$

and the input vector defined by:

$$u = \begin{pmatrix} u_d \\ u_q \end{pmatrix}. \quad (1.63)$$

The nonlinear explicit state-space representation of the PMSM motor model is given by:

$$\begin{cases} \dot{x}_1 &= nx_3x_2 - \frac{R}{L}x_1 + \frac{1}{L}u_1 \\ \dot{x}_2 &= nx_3x_1 - \frac{R}{L}x_2 + -\frac{K_t}{L}x_3 + \frac{1}{L}u_2 \\ \dot{x}_3 &= \frac{K_t}{J_m}x_2 - \frac{B}{J_m}x_3 \\ \dot{x}_4 &= x_3 \end{cases} \quad (1.64)$$

In this section, the PMSM model has first been expressed in the rotor base ( $dq$ -base) and has been expressed as a nonlinear explicit state-space system. In order to linearise this model by the differential flatness approach, the flat outputs of the model will be processed as presented in section 1.2.1.1.

#### 1.2.2.4 Flat outputs processing

In order to process the flat outputs of (1.64), it is necessary to process its implicit form given by equation (1.8) such as:

$$F(x, \dot{x}) = 0. \quad (1.65)$$

Assuming the nonlinear state-space expression of (1.64):

$$\dot{x} = f(x) + g(x)u, \quad (1.66)$$

The implicit form is then given by:

$$F(x, \dot{x}) = \begin{pmatrix} 0 \\ 0 \\ \dot{x}_3 - \frac{K_t}{J_m}x_2 + \frac{B}{J_m}x_3 \\ \dot{x}_4 - x_3 \end{pmatrix} = 0 \quad (1.67)$$

The variational system  $P(F)$  expressed in (1.20) is then given by the following matrix:

$$P(F) = \begin{pmatrix} 0 & -\frac{K_t}{J_m} & \frac{B}{J_m} + \frac{d}{dt} & 0 \\ 0 & 0 & -1 & \frac{d}{dt} \end{pmatrix} \quad (1.68)$$

According to (1.22), the Smith decomposition of  $P(F)$  is processed with :

$$S(P(F)) = (I_2, 0_{2,2}), \quad (1.69)$$

the resulting  $U \in D - Smith$  matrix equals to:

$$U = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & \frac{B + J_m \frac{d}{dt}}{K_t} & 0 & \frac{\frac{d}{dt}B + J_m \frac{d}{dt}}{K_t} \\ 0 & 1 & 0 & \frac{d}{dt} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1.70)$$

According to equation (1.25), the corresponding  $\hat{U}$  matrix satisfies:

$$\hat{U} = U \begin{pmatrix} 0_{2,2} \\ I_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{\frac{d}{dt}B + J_m \frac{d}{dt}}{K_t} \\ 0 & \frac{d}{dt} \\ 0 & 1 \end{pmatrix}. \quad (1.71)$$

As suggested in *Lemma 1.2*,  $Q \in L - \text{Smith}(\hat{U})$  is processed and is given by:

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & -\frac{d}{dt} \\ 0 & 1 & 0 & -\frac{\frac{d}{dt}B + J_m \frac{d}{dt}}{K_t} \end{pmatrix}, \quad (1.72)$$

and

$$R = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad (1.73)$$

which satisfies:

$$Q\hat{U}R = \begin{pmatrix} I_2 \\ 0_{2,2} \end{pmatrix}. \quad (1.74)$$

The following differential base  $\omega$  is obtained with equations (1.28, 1.27) :

$$\omega = \begin{pmatrix} I_2 & 0_{2,2} \end{pmatrix} Q \begin{pmatrix} dx_1 \\ dx_2 \\ dx_3 \\ dx_4 \end{pmatrix} = \begin{pmatrix} dx_1 \\ dx_4 \end{pmatrix}. \quad (1.75)$$

Hence,  $x_1 = i_d$  and  $x_4 = \theta_m$  are the flat outputs of the PMSM. According to the definition of flat outputs (1.7), the PMSM can be expressed with  $x_1 = z_1$  and  $x_4 = z_2$  and a finite number of their derivatives as shown below:

$$\left\{ \begin{array}{lcl} x_1 & = & z_1 \\ x_2 & = & \frac{J_m}{K_t} \ddot{z}_2 + \frac{B}{K_t} \dot{z}_2 \\ x_3 & = & \dot{z}_2 \\ x_4 & = & z_2 \\ u_1 & = & Rz_1 + L\dot{z}_1 - \frac{nLJ_m}{K_t} \dot{z}_2 \ddot{z}_2 - \frac{nLB}{K_t} (\dot{z}_2)^2 \\ u_2 & = & \ddot{z}_2 \frac{LJ_m}{K_t} + \ddot{z}_2 \frac{LB + RJ_m}{K_t} + \dot{z}_2 \left( \frac{RB}{K_t} + K_t \right) - nL\dot{z}_2 z_1 \end{array} \right. \quad (1.76)$$

This subsection showed the flatness of the nonlinear stepper motor model (1.64). Flat outputs  $(i_d, \theta_m)$  were identified by processing the Smith decomposition of the implicit form of the PMSM model.

In the next section, linearisation conditions of nonlinear flat systems will be discussed. Exact and pseudo- linearisation methods are presented.

### 1.2.3 Linearisation of flat systems

Linearisation methods have found many applications in the field of nonlinear system control. System linearisation is realised regarding underlying questions concerning:

- the type of linearisation: is it approximate or exact? Is the linearisation internal (input-state linearisation ) or external (input-output linearisation)?
- the goal: is there a local or aggregate linearisation required?

The approximate linearisation, around an equilibrium point, also known as quasi-linearisation, is well suited for regulation issues with a running point near from the equilibrium point. The exact linearisation is well suited for transitions of the equilibrium point and trajectory tracking. Approximate and exact linearisation often requires transformations such as static or dynamic state feedback and internal or

external decoupling.

Stable input-output decoupling by state feedback of linear systems was at first studied by P.L. Falb and W.A. Wolovitch [34] who expressed the necessary and sufficient conditions. Then, W.M. Wonham and A.S. Morse [106] introduced a pre-compensator and a dynamic feedback in order to reduce the constraints affecting stability. A major drawback of this decoupling is, for certain cases, the observability loss of a part of the state vector when realising poles-zeros simplifications in the closed loop transfer. The study of the finite zero dynamic of the transfer matrix appears to be necessary in order to reach an internal stability. A method for the realisation of a stable linear input-output decoupling was developed by W.M. Wonham [105] which consisted in replacing the static state feedback by a dynamic feedback.

Regarding nonlinear systems, A. Isidori *et al* [60] and R.M. Hirshorn [53] defined the equivalent conditions concerning the decoupling matrix regularity. This method is based on a coordinate change for the state and a static state feedback which permits to decouple and linearise inputs-outputs transfers at the same time. In the case of a non-observable state occurrence, the zero dynamic of the system, developed by C. Byrnes and A. Isidori [17] is studied. Stability criterion on the zero dynamic was then used such as exponential stability [17] or K-stability [19] to evaluate the internal stability of the looped nonlinear system. Dynamic feedback was also used for nonlinear input-output system inversion and decoupling by R.M. Hirshorn [54] and S.N. Singh [97] and [98]. Also, J. Descusse and C.H. Moog [25], M. Fliess [37] and H. Nijmeijer and W. Respondeck [87] defined nonlinear input-output systems which can be decoupled by dynamic state feedback.

### 1.2.3.1 Coordinate change and static feedback linearisation

The aim of static feedback linearisation is to determine whether a system is linear after applying a change of coordinates on the system's states and inputs. In the case of linear systems, a change of state coordinates  $x \mapsto \bar{x}$  is given with the existence

of a invertible  $M$  matrix where  $\dim(M) = n \times n$  and such as:

$$x = M\bar{x} \quad (1.77)$$

The regular static feedback  $u \mapsto \bar{u}$  is defined by an invertible  $N$  matrix of order  $m$  and a second matrix  $K$  with  $\dim(K) = m \times n$  such as :

$$u = K\bar{x} + N\bar{u} \quad (1.78)$$

The global static feedback is given by the following system:

$$\begin{pmatrix} \bar{x} \\ \bar{u} \end{pmatrix} \mapsto \begin{pmatrix} x \\ u \end{pmatrix} = \begin{pmatrix} M & 0 \\ K & N \end{pmatrix} \begin{pmatrix} \bar{x} \\ \bar{u} \end{pmatrix} \quad (1.79)$$

Assuming the following system:

$$\dot{x} = f(x, u), x \in \mathbb{R}^n, \quad u \in \mathbb{R}^m \quad (1.80)$$

Linearisation of system (1.80) by static feedback consists in finding a state vector  $z$  and an input vector  $v$  such that (1.80) is equivalent to the following linear system :

$$\dot{z} = Az + Bv \quad (1.81)$$

The generalised form of (1.79) is given by the nonlinear transformations :

$$\begin{pmatrix} \bar{x} \\ \bar{u} \end{pmatrix} \mapsto \begin{pmatrix} z = \phi(x) \\ v = k(x, u) \end{pmatrix}, \quad (1.82)$$

where  $\phi$  is a smooth mapping. It was shown by Charlet *et al* [20] that dynamic feedback is useful only in the case of multi-input systems linearisation. Single-



inputs systems are therefore considered such as :

$$\dot{x} = f(x) + g(x)u \quad , \text{ with } x \in \mathbb{R}^n, u \in \mathbb{R}^m. \quad (1.83)$$

In the case of single-inputs, system (1.83) is equivalent to a controllable linear system by static feedback and diffeomorphism if the two following conditions are satisfied :

- $G_{n-2}$  has a constant rank and is involute on a neighbourhood  $V$  of the origin,
- the rank of  $G_{n-1}$  is  $n$ ,

where the distribution of vectors fields  $G_i$  is given by :

$$G_i = sp\{g, ad_f g, \dots, ad_f^i g\} \quad (1.84)$$

with  $ad_f^i g$  the Lie bracket of  $f$  and  $g$  repeated  $i$  times and  $ad_f^i = [f, ad_f^{i-1}]$ .

When the system satisfies the two previous conditions, it can be linearised by the given feedback and diffeomorphism:

$$u = \alpha(x) + \beta(x)v, \quad (1.85)$$

$$\xi = \phi(x) \quad (1.86)$$

Determining the feedback and diffeomorphism consists of processing in a neighbourhood of  $V$  the following partial derivatives system:

$$\left\{ \begin{array}{lcl} L_g \phi_1 & = & 0 \\ L_{ad_f g} \phi_1 & = & 0 \\ \vdots & & \\ L_{ad_f^{n-2} g} \phi_1 & = & 0 \end{array} \right. \quad (1.87)$$

where  $\beta(x)$  is a square invertible matrix.  $\alpha$  and  $\beta$  are given by:

$$\begin{cases} \alpha = -\frac{L_f^n \phi_1}{L_g L_f^n \phi_1} \\ \beta = \frac{1}{L_g L_f^n \phi_1} \end{cases}, \quad \phi_i = L_f^{i-1} \phi_1, \quad \forall i = 2, \dots, n \quad (1.88)$$

where  $L_f g(x)$  is the lie derivative of the smooth function  $(g)$  along the vector field  $f$ . Its expression is given by

$$L_f g(x) = \sum_{i=1}^n f_i(x) \frac{\partial}{\partial x_i} g(x) \quad (1.89)$$

and  $u = k(z, v)$ .

**Remark 1.2.** *Only systems with linear inputs were considered in this case because it corresponds to the system hypothesis established by Jakubczyk-Respondek and Hunt-Su-Meyer. This result was generalised for nonlinear inputs systems [62] by setting the distributions  $G_i$  to:*

$$G_0 = sp \left\{ \frac{\partial f}{\partial u} \right\} \quad (1.90)$$

$$G_i = G_{i-1} + ad_f G_{i-1}, \quad \text{for } i \geq 1. \quad (1.91)$$

**Definition 1.6. (Brunovsky form)** *A linear controllable system is equivalent after static feedback and a change of base of its coordinates, to its Brunovsky form:*

$$\begin{cases} y^{(k_1)} = v_1 \\ \vdots \\ y_m^{(k_m)} = v_m \end{cases} \quad (1.92)$$

where  $k_1, \dots, k_m$  are the system controllability subscripts and  $v_1, \dots, v_m$  are the inputs of the equivalent linear system. Each input  $v$  equals to the  $k$ -derivative of its corresponding output.

After diffeomorphism (see Appendix A.2) and static feedback, system (1.80) is equivalent to a system containing a linear part and a nonlinear transformation given

by:

$$\begin{cases} \dot{z} = Az + Bv \\ \dot{\xi} = a(z, \xi) + b(z, \xi)v \end{cases} \quad (1.93)$$

When a nonlinear system is not compliant with the conditions of the Respondek and Hunt-Meyer theorem [63], the system is not locally linearisable but several partial linearisations are possible. R. Marino [79] specified the size of the largest linearisable subsystem. Stabilisation is therefore only possible for the linearisable part. The behaviour of the non-linearisable part is moreover unknown and it is only possible to remark the stability or instability of the system after feedback. Considering this theorem, one might wonder if a partial static feedback linearisable system is differentially flat. In the case of partially static feedback linearisable single-input systems, B.Charlet *et al* showed that the extension to the dynamic feedback brings no benefits to the linearisation problem. Such systems are therefore not linearisable and constitute non flat systems. On the other hand, when multi-input systems are considered, it is possible to linearise the input-state behaviour with an endogenous dynamic extension realised by a dynamic state feedback as presented in the next section. The system is linearisable regarding the state and the linearisation can be processed on the equivalent system. If a multiple-input system is not dynamic feedback linearisable it is not flat.

### 1.2.3.2 Endogenous dynamic feedback linearisation

Consider system (1.80). Dynamic feedback is given by a differential equation also called dynamical compensator given by equation (1.94),

$$\dot{z} = \beta(x, z, v) \quad (1.94)$$

and a feedback loop:

$$u = \alpha(x, z, v). \quad (1.95)$$

The dynamic feedback is now expressed with (1.80, 1.93 and 1.94) by the following system:

$$\begin{cases} \dot{x} = f(x, \alpha(x, z, v)) \\ \dot{z} = \beta(x, z, v). \end{cases} \quad (1.96)$$

System (1.80) is linearisable by dynamic feedback if system (1.96) is linearisable by static feedback. If the linearised system is also L-B equivalent [41], it is an endogenous feedback linearisation, meaning a Lie-Bäcklund isomorphism  $\phi$  and its inverse  $\psi$  exist such as:

$$(x, \bar{u}) = \phi(x, z, \bar{v}), \quad \bar{u} = (u, \dot{u}, \ddot{u}, \dots, u^{(m)}) \quad (1.97)$$

and

$$(x, z, \bar{v}) = \psi(x, \bar{u}), \quad \bar{v} = (v, \dot{v}, \ddot{v}, \dots, v^{(m)}), \quad (1.98)$$

which implies that  $z, v, \dot{v}, \dots$ , may be expressed as a function of  $x, u$  and a finite amount of derivatives of  $u$ .

**Remark 1.3.** *Every nonlinear flat system is endogenous dynamic feedback linearisable and every endogenous dynamic feedback linearisable system is flat [73].*

The previous linearisation methods are exact and allow, if linearisability conditions are satisfied, the expression of an equivalent system independently of any equilibrium point such as the tangent linearisation. The next section presents the case of pseudo-linearisation, where not only the dynamic must be invariant along the equilibrium set but also the whole linear tangent model must be fixed, with respect to an appropriate coordinate frame. A pseudo-linearised system is still nonlinear but its non linearities are of the first-order around any equilibrium point.

### 1.2.3.3 Pseudo-linearisation

Current linearisation methods such as tangent linearisation, allow definition of an equivalent linear system in the neighbourhood of an equilibrium point of the system.

The resulting linear system is then controllable only around this point. P. Mouyon described [45] a linearisation method for multiple-input nonlinear systems, through approximate linearisation of the state equations. The aim is to obtain invariance along the equilibrium set of the whole linear tangent model independently of the system's poles. Therefore, considering the following coordinate change:

$$z = T(x) \quad (1.99)$$

and the state feedback:

$$v = S(x, u) \quad (1.100)$$

The linear tangent model of the closed loop system which shall be independent of the equilibrium point an is given by:

$$\delta\dot{z} = A\delta z + B\delta v, \quad (1.101)$$

where  $A, B$  are constant matrices. The pseudo-linearised system is nonlinear but its non linearities are of the first-order around any equilibrium point. The pseudo-linearised system is given by:

$$\dot{z} = A\delta z + B\delta v + \varepsilon(z, v), \quad (1.102)$$

such as  $\varepsilon(z_0, v_0) = 0$  and  $d\varepsilon(z_0, v_0) = 0$ . In the case of multi-inputs nonlinear systems, projections of the partial derivative equations have to be solved in the tangent space  $\mathcal{V}$  and its orthogonal. This is why integration on the tangent space requires an involution criterion regarding the projected fields. Considering the following tangent model at the equilibrium set  $(x_0, u_0)$ :

$$\delta\dot{x} = F(x_0)\delta x + G(x_0)\delta u. \quad (1.103)$$

It has been shown that a system is pseudo-linearisable if and only if the distribution of the following vector fields:

$$\{G, FG, \dots, F^{k_i-2}G\} \cap T\mathcal{V} \quad (1.104)$$

are involutive, where  $T\mathcal{V}$  is the tangent space to  $\mathcal{V}$  and  $k_i$  are the controllability indices of  $(F, G)$ . The rank of this distribution is given by :

$$\text{rank} \left[ \{G, FG, \dots, F^{k_i-2}G\} \cap T\mathcal{V} \right] = \text{card}\{j/k_j < k_i\}. \quad (1.105)$$

A controllability canonical form is chosen such as :

$$\begin{cases} \delta \dot{z}_1 &= \delta z_2 \\ \vdots & \\ \delta \dot{z}_{n-1} &= \delta z_n \\ \delta \dot{z}_n &= \delta v \end{cases} \quad (1.106)$$

which implies according to (1.103, 1.99 and 1.100) :

$$\begin{cases} \frac{\partial T_1}{\partial x}(F\delta x + G\delta u) &= \frac{\partial T_2}{\partial x}\delta x \\ \vdots & \\ \frac{\partial T_{n-1}}{\partial x}(F\delta x + G\delta u) &= \frac{\partial T_n}{\partial x}\delta x \\ \frac{\partial T_n}{\partial x}(F\delta x + G\delta u) &= \frac{\partial S}{\partial x}\delta x + \frac{\partial S}{\partial u}\delta u. \end{cases} \quad (1.107)$$

System (1.107) must be true independently of  $\delta x$  and  $\delta u$ , therefore:

$$\left\{ \begin{array}{l} \frac{\partial T_2}{\partial x} = \frac{\partial T_1}{\partial x} F \\ \vdots \\ \frac{\partial T_n}{\partial x} = \frac{\partial T_{n-1}}{\partial x} F \\ \frac{\partial S}{\partial x} = \frac{\partial T_n}{\partial x} F \end{array} \right. \quad \text{and} \quad \left\{ \begin{array}{l} \frac{\partial T_1}{\partial x} G = 0 \\ \vdots \\ \frac{\partial T_{n-1}}{\partial x} G = 0 \\ \frac{\partial T_n}{\partial x} G = \frac{\partial S}{\partial u}. \end{array} \right. \quad (1.108)$$

The previous set of  $n - 1$  first-order homogeneous equations satisfies for  $T_1$  at the equilibrium point  $x_0$ :

$$\frac{\partial T_1}{\partial x} [G(x_0), F(x_0)G(x_0), \dots, F^{n-2}(x_0)G(x_0)] = 0. \quad (1.109)$$

The other equations of the  $T_i$  gradients are given at the equilibrium point by:

$$\frac{\partial T_i}{\partial x} = \frac{\partial T_1}{\partial x} F^{n-1}(x_0), \quad i = 2, \dots, n \quad (1.110)$$

and the gradient of  $S$  by:

$$\left( \frac{\partial S}{\partial x}, \frac{\partial S}{\partial u} \right) = \frac{\partial T_1}{\partial x} F^{n-1}(x_0) (F(x_0), G(x_0)) \quad (1.111)$$

$T$  and  $S$  are then found by integration of (1.110) and (1.111).

In this section static feedback and dynamic feedback linearisation methods were presented. Linearisability conditions must be satisfied and are not always easy to demonstrate, in particular for multiple-input nonlinear systems. If such systems are flat, it was previously shown that they are endogenous feedback linearisable and therefore an equivalent linear system can be determined by finding an appropriate feedback and a smooth mapping.

Endogenous feedback linearisation is applied in the next section to the flat HSM model given in equation (1.112). It is shown that the system is static feedback linearisable and can be expressed with an equivalent linear system given by the Brunovsky form.

### 1.2.3.4 Static feedback linearisation of a hybrid stepper motor

The flat PMSM model was given by the following equations where the states and the inputs are function of the flat outputs  $z_1 = i_d$  and  $z_2 = \theta_m$  and their derivatives such as:

$$\left\{ \begin{array}{lcl} x_1 & = & z_1 \\ x_2 & = & \frac{J_m}{K_t} \ddot{z}_2 + \frac{B}{K_t} \dot{z}_2 \\ x_3 & = & \dot{z}_2 \\ x_4 & = & z_2 \\ u_1 & = & Rz_1 + L\dot{z}_1 - \frac{nLJ_m}{K_t} \dot{z}_2 \ddot{z}_2 - \frac{nLB}{K_t} (\dot{z}_2)^2 \\ u_2 & = & \ddot{z}_2 \frac{LJ_m}{K_t} + \ddot{z}_2 \frac{LB + RJ_m}{K_t} + \dot{z}_2 \left( \frac{RB}{K_t} + K_t \right) - nL\dot{z}_2 z_1 \end{array} \right. \quad (1.112)$$

According to equation (1.85) and with  $\beta(x)$  invertible:

$$v = \beta^{-1}(x) [u - \alpha(x)]. \quad (1.113)$$

Also,

$$v = \beta^{-1}(x)u - \beta^{-1}(x)\alpha(x). \quad (1.114)$$

By setting  $\Delta(x) = \beta^{-1}(x)$  and  $\Delta_0(x) = -\beta^{-1}(x)\alpha(x)$ , (1.114) becomes:

$$\Delta_0(x) + \Delta(x)u = v. \quad (1.115)$$

According to the Brunovsky form, equation (1.115) equals to:

$$\Delta_0(x) + \Delta(x) \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} z_1^{(\rho_1)} \\ z_2^{(\rho_2)} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (1.116)$$



where  $\rho_1 + \rho_2 = n$ . Equation (1.115) implies:

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \Delta^{-1}(x) \left( \begin{pmatrix} z_1^{(1)} \\ z_2^{(3)} \end{pmatrix} - \Delta_0(x) \right) \quad (1.117)$$

with:

$$\Delta_0(x) = \begin{pmatrix} \Delta_{01}(x) \\ \Delta_{02}(x) \end{pmatrix} \quad (1.118)$$

where:

$$\Delta_0(x) = \begin{pmatrix} -\frac{R}{L}x_1 + nx_3x_2 \\ -\frac{K_t(BL + RJ_m)}{LJ_m^2}x_2 + \frac{B^2L - J_mK_t^2}{LJ_m^2}x_3 + \frac{nK_t}{J_m}x_3x_1 \end{pmatrix} \quad (1.119)$$

and

$$\Delta(x) = \begin{pmatrix} \frac{1}{L} & 0 \\ 0 & \frac{K_t}{LJ_m} \end{pmatrix} \quad (1.120)$$

According to equation (1.116) the equivalent linear system is given by:

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{s} & 0 \\ 0 & \frac{1}{s^3} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (1.121)$$

The presented flat HSM model is given by a linear Brunovsky form. It may be observed that the dimensions of flat output  $z$  and input vector  $v$  are the same, as explained for static feedback linearisation in the previous paragraph. It is shown in the next section that endogenous feedback linearisation properties have interesting applications in model-based fault detection and diagnosis (FDD & FDI). Indeed, model-based diagnosis approaches of nonlinear observers such as the extended Kalman Filter (EKF) use approximated linearisation and may not always meet the required stability and performance conditions. Linearising the model by exact linearisation before synthesising the observer might increase the performance

level of the system and reduce processing costs of the computer unit.

In the next section, a state of the art in current analytical diagnosis methods is described, particularly addressing residual analysis methods for aircraft control systems such as actuators and sensors. Among fault detection and isolation (FDI) methods, linear and nonlinear observers will be detailed [82, 72].

### 1.3 Model-based monitoring

The development of aircraft control systems such as electric engine computer units (EECU), flight computer units (FCU) or actuators and sensors are designed regarding high performance requirements.

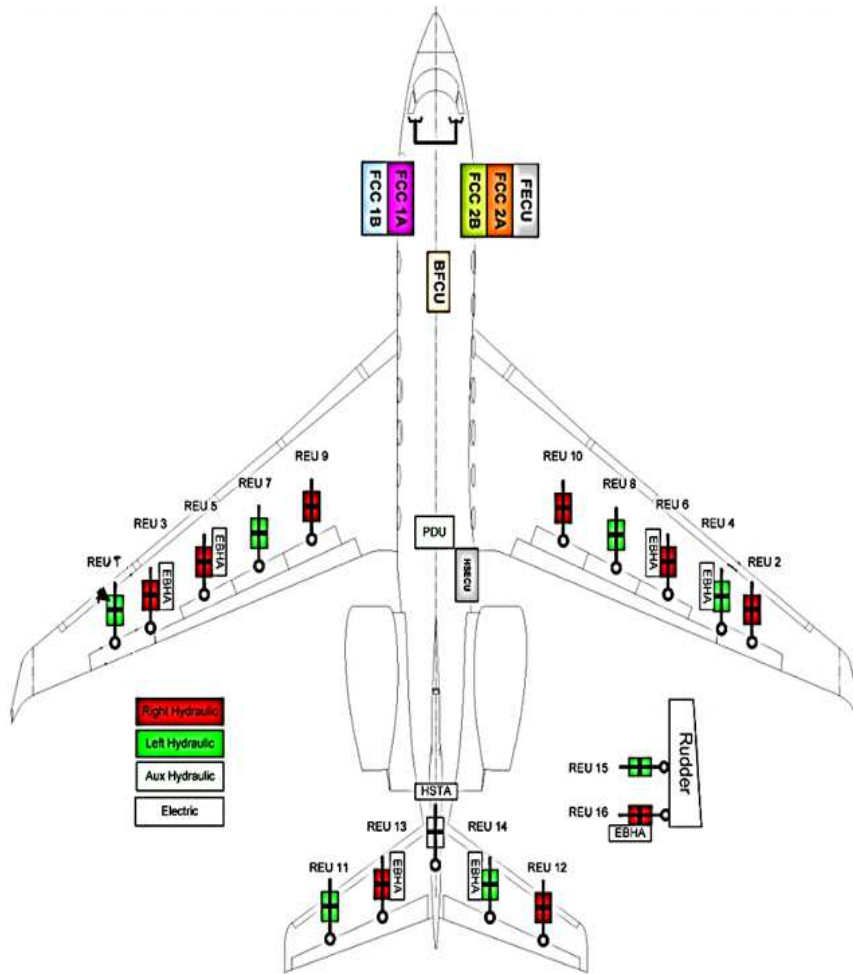


Figure 1.6: Flight control actuator redundancies

Reaching these goals led to an increase of complexity qualifying such systems as high-integrity systems (HIS). HIS need to be as much fail-safe and fault-tolerant as possible, implying on one hand physical redundancies which allow to ensure the availability and integrity of a given system through the multiplication of rows of actuators, sensors or computer units. Therefore, critical functions such as: the motion of flaps or the velocity and altitude measurement are secured. The depicted hydraulic actuators in figure 1.6 are controlled by redundant remote electronic units (REU). On the other hand, embedded monitoring systems must be able to detect, isolate and identify any fault occurrence [50] where:

- *detection*, consist in making a binary decision: a fault occurred on the system or not,
- *localisation*, is the ability to determine the defect component,
- *identification*, is the ability of the system to process the fault behaviour in order to reconfigure the system after the fault occurred. In this case, a fault-model is required.

In most current monitoring systems, only the two first steps are designed. These algorithms are denoted as fault detection and isolation (FDI) algorithms. FDI methods can be split into two main approaches: model-based and data based approaches (Figure 1.7), presenting for each subset qualitative and quantitative methods.

According to linearisability properties of the flat HSM model shown in section 1.2, a quantitative model-based approach such as the observer-based FDI method was chosen to perform states estimations for this study. The description of other approaches shown in figure 1.7 can be found in [58, 26, 47]. Model-based monitoring appeared in the early 1970's and has since found much practical application in the oil, aerospace, automotive, nautical and rail industries. The aim of model-based diagnosis [57, 26, 102] is to allow detection of fault occurrences and to identify their origin using fault detection, isolation and diagnosis algorithms (FDD & FDI). For reasons of space weight and cost (SWaP), multiplying physical components is not

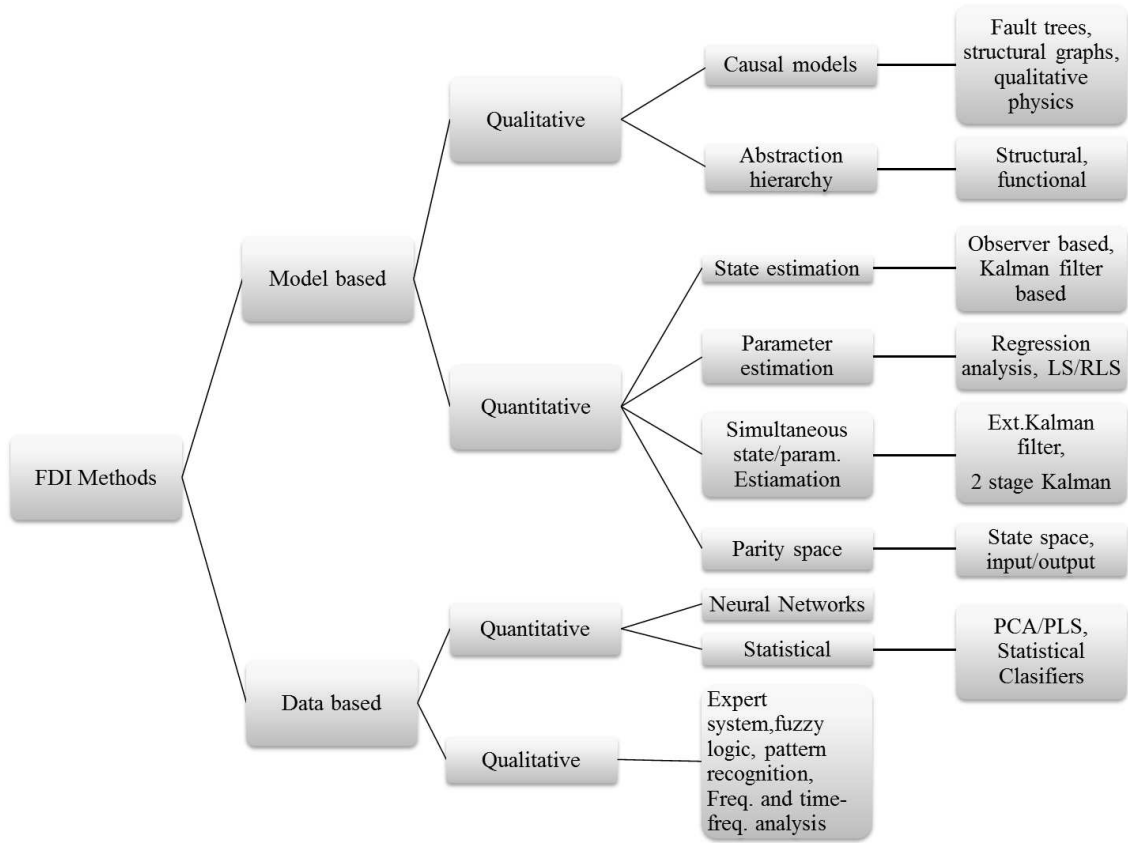


Figure 1.7: Classification of fault-detection and isolation (FDI) methods

always possible. Therefore, modern monitoring units require analytic redundancy (Figure 1.8). Analytic redundancy, also called software redundancy in computer

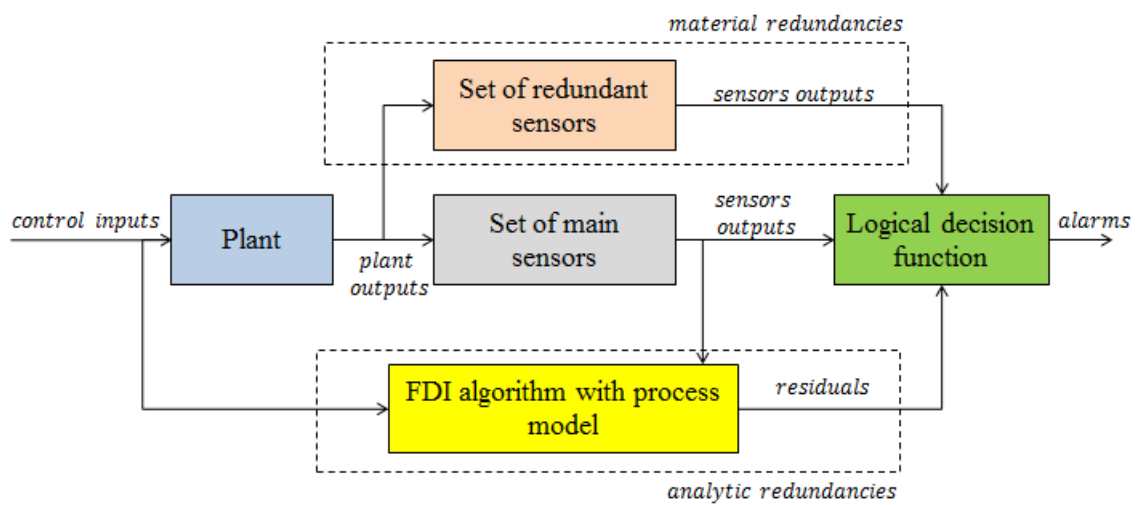


Figure 1.8: System monitoring scheme based on material and analytic redundancies

units, can be used as a complement to physical redundancy. Dynamic models are used to process estimates of measured variables. They are fed with the same inputs

as the physical system and generate as a result an estimation of the system's states. The type of model-based approach that is chosen here is a residual generator as shown in figure 1.9. A decision logic is required to detect and generate an alert in the case of an actuator or sensor fault occurrence. Generally, decision logics allow to detect faults when the residuals exceeds a certain threshold. Fault isolation is then required in case of multiple fault cases to determine the type of fault. Residuals must be both fault-sensitive, detecting each occurrence of a fault (performance goal) and insensitive to noise (robustness goal). FDI systems are also designed regarding the trade-off between false-alarm and non-detection rates (FAR & FDR).<sup>5</sup> Among

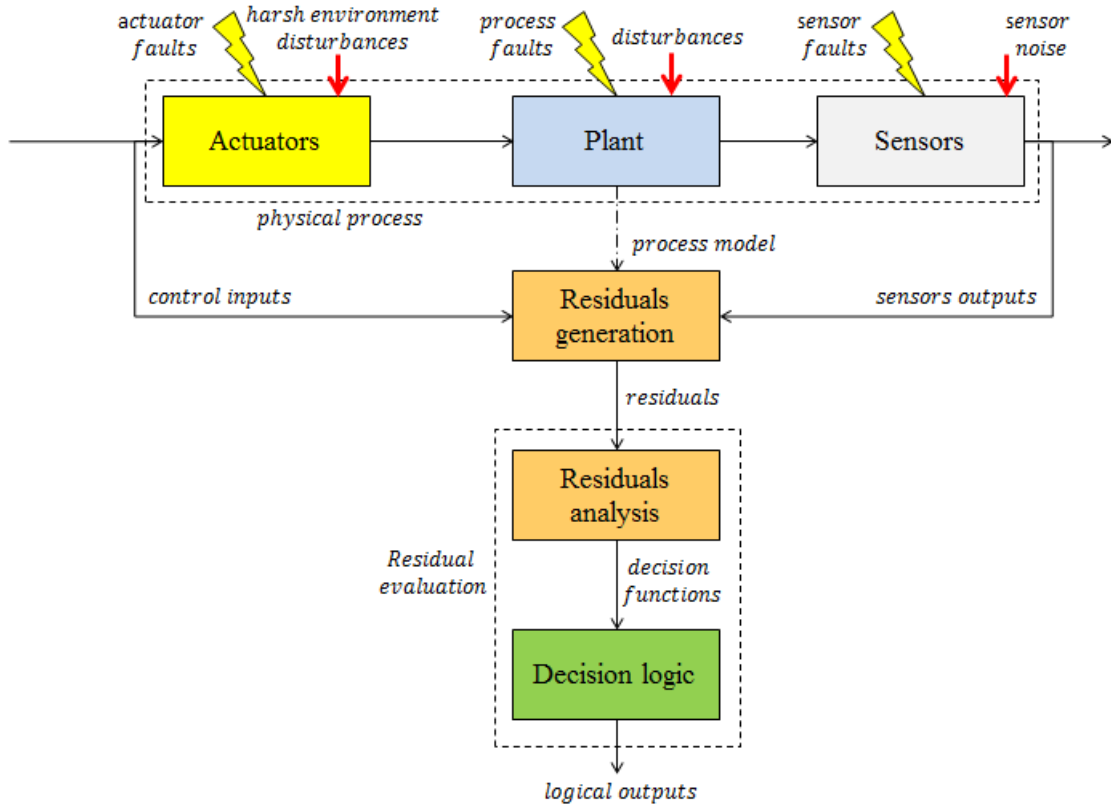


Figure 1.9: Model-based fault-detection schematic

the depicted analytical fault-detection and isolation methods, the focus here will be on observer-based methods which are interesting for flight-critical systems.

<sup>5</sup>FAR designs the rate of alarm generations by the FDI system when no fault occurred

### 1.3.1 Residual generation

Among FDI methodologies, one of the most attractive is known as the Fundamental Problem of Residual Generation (FPRG). Each residual is then made sensitive only to a single fault thus ensuring fault isolation in the case of multiple faults.

#### 1.3.1.1 The Fundamental Problem of Residual Generation (FPRG)

The following nonlinear control-affine system is considered in this case [84] with:

$$\begin{cases} \dot{x} = f(x) + \sum_{i=1}^m g_i(x)u_i + \sum_{i=1}^{m_\psi} l_i(x)\psi_i + \sum_{i=1}^{m_d} p_i(x)v_i \\ y = h(x) \end{cases} \quad (1.122)$$

where  $\psi_i$  is the faults vector and  $v_i$  the disturbances vector and the  $l$  and  $p$  functions are smooth manifolds. The FPRG consists in finding a filter such that the following system:

$$\begin{cases} \begin{pmatrix} \dot{x} \\ \dot{\zeta} \end{pmatrix} = \begin{pmatrix} f(x) \\ \bar{f}(y, \zeta) \end{pmatrix} + \sum_{i=1}^m \begin{pmatrix} g_i(x) \\ \bar{g}_i(y, \zeta) \end{pmatrix} u_i + \sum_{i=1}^{m_\psi} \begin{pmatrix} l_i(x) \\ 0 \end{pmatrix} \psi_i + \sum_{i=1}^{m_d} \begin{pmatrix} p_i(x) \\ 0 \end{pmatrix} v_i \\ e = \bar{h}(y, \xi) = h^e(x^e) \end{cases} \quad (1.123)$$

exists in the neighbourhood of the origin  $x^e$ , then the following properties are satisfied:

1. if  $\psi = 0$ , then the residual  $e$  is not affected by  $u_i$  and  $v_j$ ,
2.  $e$  is affected by  $\psi$ ,
3.  $\lim \|e(t, x_0, \zeta_0, u, \psi = 0, v)\| = 0$ , if there is no faults, the residual  $e$  converges to zero for any initial set  $(x_0, \zeta_0)$  chosen in a mapping containing the origin  $(x, \zeta)^T = (0, 0)^T$  and for all acceptable inputs, where,  $\zeta \in \mathbb{R}^q, 1 \leq q \leq n$  and  $e \in \mathbb{R}^s, 1 \leq s \leq p$ .

**Definition 1.7. (Fault sensitivity of the residual [22])** A residual  $r(t, u, y, v)$  is not affected by  $v \in V$  if for any input  $u$  and output  $y$ ,  $r(t, u, y, v_1) = r(t, u, y, v_2)$ ,  $\forall t \geq 0, \forall \{v_1; v_2\} \in V^2$ . Else, if this condition is not satisfied,  $r$  is affected by  $v$ .

Three main approaches to residual generation have been identified :

- *Parameter estimation*, where the residual quantifies the difference between the real parameter value and the model reconstructed parameter.
- *State estimation*, in which observers dedicated to linear and nonlinear systems are used for state estimation. The error between estimate and the output measure is the residual signal. The Kalman filters are used with stochastic signals.
- *Static parity space* [51], in which only the output equation of a state space model is considered. The parity matrix  $W$  is processed such as  $r = WCx = Wy$ .  $W$  is chosen for compliance with the robustness and performance goals required of the residual.

In order to improve the fault sensitivity of the residual, faults can be generated and become part of the plant model as described in the next paragraph.

### 1.3.1.2 Fault modelling

There exist several ways to model faults, among them the following system extension such as:

$$\begin{cases} \dot{x} &= Ax + Bu + E_d d + E_f f \\ y &= Cx + Du + F_d d + F_f f \end{cases} \quad (1.124)$$

where  $d$  is the disturbance vector and the  $E_d$  matrix indicates which input is affected by the disturbancy. The  $f$  vector is an unknown vector that represents all possible faults and will be zero in the fault-free case. The  $E_f$  matrix indicates where the fault occurs. Faults are divided into three categories:

- *sensor faults*: these are faults that directly act on the process measurement

- *actuator faults*: these faults cause changes in the actuator
- *process faults*: they are used to indicate malfunctions within the process.

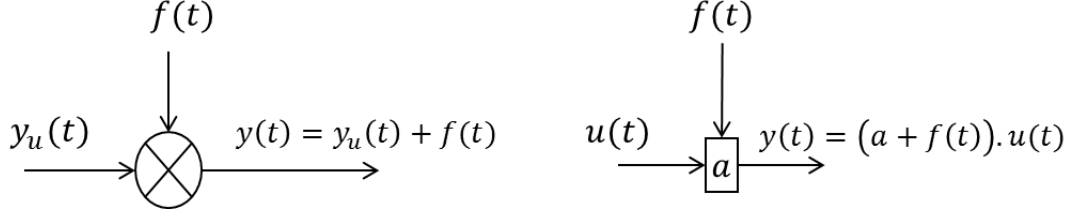


Figure 1.10: Additive and multiplicative faults

These faults can be seen as additive or multiplicative as shown in figure 1.10, where  $y_u(t)$  is an input signal without fault and  $y(t)$  is the defect signal affected by the  $f(t)$  fault. In the case of multiplicative faults,  $a$  is a time invariant model parameter (it can be a coil resistance for example) and  $u$  is an input signal not affected by the fault. Fault modelling in a model-based diagnosis approach are based on the following definitions:

**Definition 1.8. (Fault)** *A fault is an unpermitted deviation of at least one characteristic property(feature) of the system from the acceptable, usual, standard condition.*

**Definition 1.9. (Failure)** *A failure is a permanent interruption of a system's ability to perform a required function under specified operating conditions.*

**Definition 1.10. (Malfunction)** *A malfunction is an intermittent irregularity in the fulfilment of a system's desired function.*

The link between fault, failure and malfunction is shown in figure 1.11. After adding fault matrices in the system model, a decision has to be made regarding certain requirements such as FAR and NDR or robustness and performance of the FDI algorithm. The next subsection presents current fault-detection methodologies.

### 1.3.2 Fault detection

After generating the residual signals carrying the fault information, a decision concerning the system's health level has to be made. This boolean output is generated



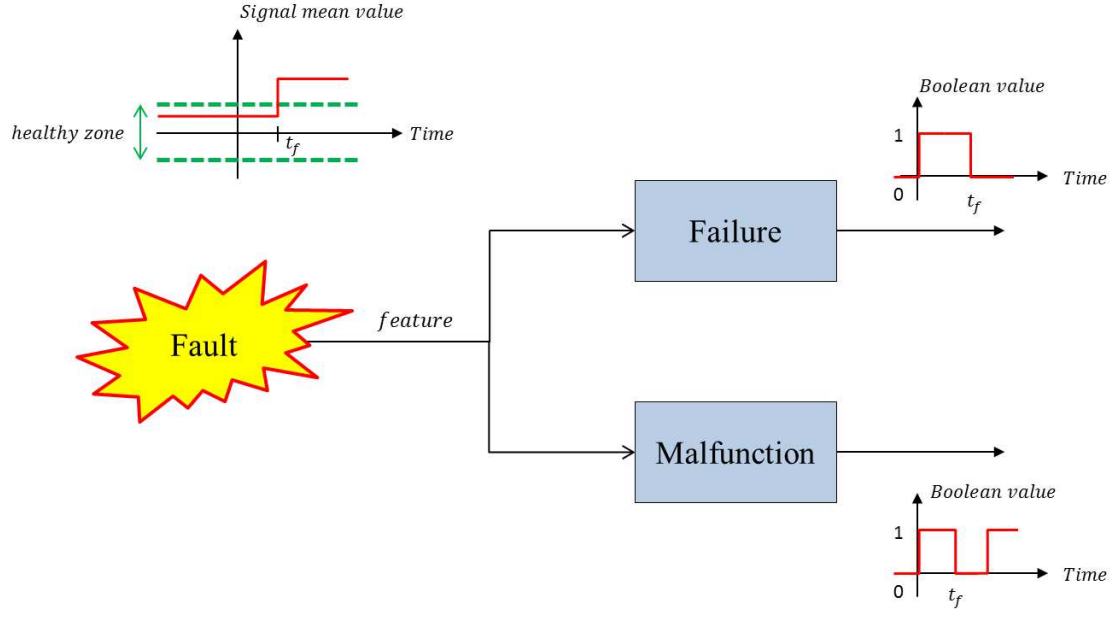


Figure 1.11: Consequences of a fault occurrence

by residual evaluation functions [3]. Model-based fault detection requires high-fidelity modelling of the dynamics of the system, and generation of a residual signal as fault sensitive as possible and insensitive to noise and disturbance. In a safe mode, the residual is a stochastic signal and its mean value is close to zero. When a fault occurs the signal mean value varies on the fault event, as shown in figure (1.12). Statistical methods, such as normal distributions  $N(\mu, E)$ , are used in order to process the signals average  $\mu$  and its covariance matrix  $E$ . A simple way to detect a fault is to generate an alarm when the residual exceeds a threshold, defined within the systems specifications. As an example, let  $\mu_0$  and  $\mu_1$  correspond respectively to the residual average without faults and with fault. Hypotheses  $H_0$  and  $H_1$  are such as:

$$\begin{cases} H_0 & : & \mu & = & \mu_0 \\ H_1 & : & \mu & = & \mu_1 \end{cases} \quad (1.125)$$

If  $H_1$  is chosen while  $H_0$  is true, the decision is a false alarm.

If  $H_0$  is chosen while  $H_1$  is true, the decision is a non-detection.

It is important to note that the residual  $r$  may not always have a Gaussian distribution [71].

	$H_0$ decision	$H_1$ decision
$H_0$ true (no fault)	right decision	false alarm
$H_1$ true (fault)	no detection	right decision

Table 1.1: False alarm and non-detection hypothesis

A local representation of the residual can be given by:

$$r_{loc} = \frac{1}{\sqrt{N}} \sum_{t=1}^N r(t) \quad (1.126)$$

where  $r_{loc}$  is considered as a Gaussian signal only if  $N$  is great enough. Other statistical based fault detection methods could be listed such as *fixed threshold*, *the student test*, *the generalized likelihood ratio test (GLR)*, *sequential probability ratio test (SPRT)*, *the CUSUM test* and *Randomized Sub-Sampling (RSS)*

### 1.3.3 Fault isolation

The final step of residual evaluation involves isolation of the faulty residual. Considering that several residuals are generated and each residual is processed to be

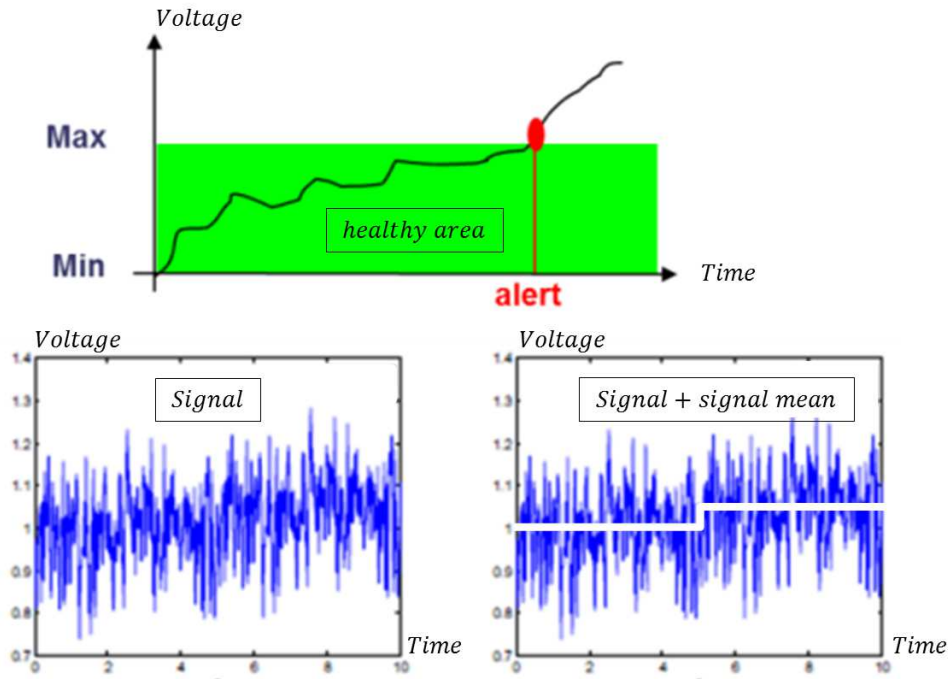


Figure 1.12: Signal mean change on fault occurrence

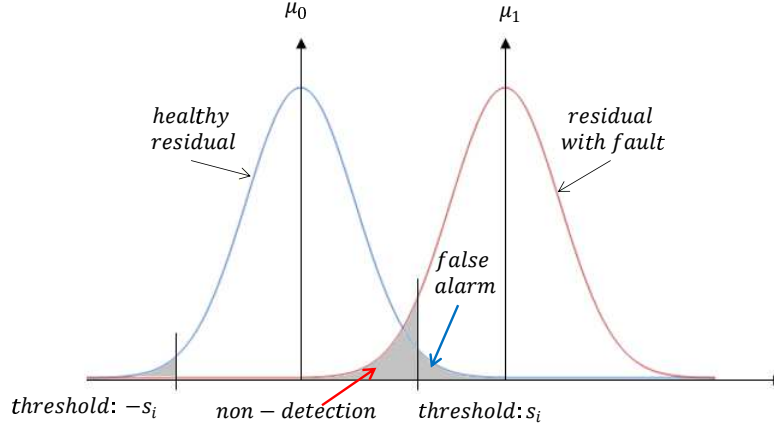


Figure 1.13: Probability density of a variable and mean change detection

sensitive to one particular fault only. Once a fault occurred, many residuals are sensitive to a same fault. Therefore, one residual is not sufficient to isolate the source of the fault. A vector of residuals is generated and sensitive to several faults. If a variation occurs on more than one residual at the same time, fault tables allow to isolate the fault. It is then possible to determine if the fault is related to a sensor or an actuator. Terms of "sensor faults" and "actuator faults" are used in fault-isolation. Two main methods are described here:

- Residual structures
- Directional residuals

In the case of residual structures [83], a binary table is used to reflect the fault affected to each of three residuals as shown in Table (1.2). When the  $i^{th}$  residual is sensitive to the  $j^{th}$  fault, then the value "1" is placed to the corresponding cell. Placing "0" means that the residual is not sensitive to the corresponding fault. Table

	$f_1$	$f_2$	$f_3$
$r_1$	1	0	0
$r_2$	0	1	0
$r_3$	0	0	1

a)

	$f_1$	$f_2$	$f_3$
$r_1$	1	1	0
$r_2$	1	0	1
$r_3$	0	1	1

b)

Table 1.2: Fault signatures on residual structures

(1.2) shows an isolability rank of two because two digits need to be changed to go

from one fault vector to another. Both tables have the same isolability rank but table  $a$ ) contains more "0". Therefore, its configuration is preferred.

**Definition 1.11. (Fault localisation)** *A fault is structurally localisable if all the columns of the fault table are different.*

### 1.3.3.1 Parity space

In the case of directional residuals, the residual vector  $r$  is collinear to the fault vector  $W_{rf}^{[i]}$ . The parity matrix  $W$  is processed such as  $r = WCx = Wy$ .  $W$  is chosen for compliance with the robustness and performance goals required of the residual. For static parity space, the residual is generated with the given model:

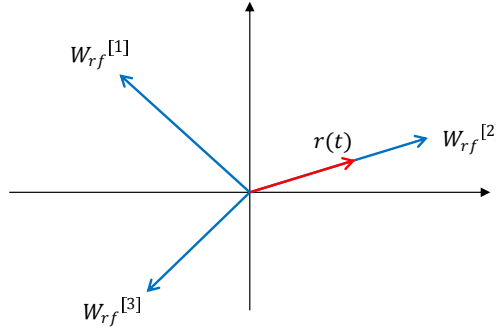


Figure 1.14: Directional residual collinear to a fault vector  $f_2$

$$y(t) = Cx(t) + D_d d(t) + D_f f(t). \quad (1.127)$$

Hence the residual  $r(t)$  given by:

$$r(t) = Wy(t) = WCx(t) + WD_d d(t) + WD_f f(t), \quad r(t) \in \mathbb{R}^{p-n} \quad (1.128)$$

with,

$$f(t) = \begin{bmatrix} f_1 \\ \vdots \\ f_{m_f}(t) \end{bmatrix}, \quad f \in \mathbb{R}^{m_f}. \quad (1.129)$$

The parity matrix  $W \in \mathbb{R}^{(p-n) \times n}$  is chosen such as the residual is insensitive to disturbances  $d(t)$  and to the state which implies  $WC = 0$ ,  $WD_d d(t) = 0$  and

$W_{rf} = WD_f \in \mathbb{R}^{(p-n) \times m_f}$ . The parity vectors sensitive to faults are then given by:

$$W_{rf} = \begin{bmatrix} W_{rf}^{[1]} & \dots & W_{rf}^{[m_f]} \end{bmatrix} \quad (1.130)$$

In figure (1.14), the isolated fault is  $f_2$  because the residual is collinear to  $W_{rf}^{[2]}$ .

The next subsection deals with the application of residual generation methods to linear and non-linear observers. System controllability and observability fundamentals are recalled in order to understand observers processing.

### 1.3.4 Linear observers

System monitoring using state-space representations consists in studying the coherency of the model behaviour regarding the real system. One of these approaches is based on the comparison of measured variables and the on-line processed variables from the model, while the physical system and the model are have the same inputs. To process the outputs, it is necessary to know certain state-variables. A first step in the computing model outputs is dedicated to the estimation of these unknown states. The system which permits this processing is called *observer*. An observer is defined as a dynamical system which takes as an input the known signals of the physical system and which outputs converge to an estimation of the state variables, or a subset of the state variables if certain states are not observable. Observability definitions are recalled in the next subsection.

#### 1.3.4.1 Observability

The following definitions are based on the work of Gauthier and Bornard as well as Hermann and Krener [48, 52].

**Definition 1.12. (Observability)** *The observability of a system is realised on its fault-free model given by:*

$$\begin{cases} \dot{x} &= f(x, u) \\ y &= h(x, u). \end{cases} \quad (1.131)$$

System (1.131) is observable if any state is distinguishable from any other state.

**Definition 1.13. (Indistinguishability)** Two initial states are indistinguishable if for any input, each state dedicated trajectory generates the same output.

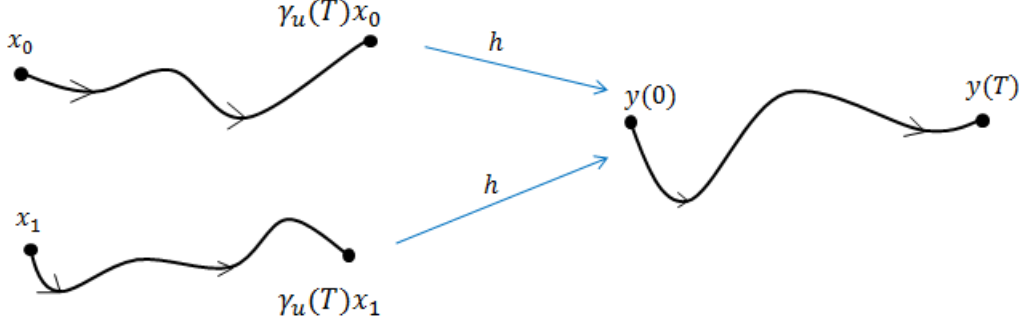


Figure 1.15: Indistinguishable states

Also, for a given state  $x(t_0)$  and a given input  $u(t_0)$ , there can only be one output  $y(t_0, t_1)$ . To consider global observability, all the states of system (1.66) must be distinguishable. The definition of  $U$ -indistinguishability defines local observability.

**Definition 1.14. (Local indistinguishability)** Let  $U$  be a subset of  $X \subseteq \mathbb{R}^n$  with two initial states  $x_1$  and  $x_2$ . The two states are  $U$ -Indistinguishable if  $\forall t \geq t_0$ , the corresponding outputs  $y_1$  and  $y_2$  are identical for any bounded measurable control  $t \mapsto u(t)$  and if,  $\forall t \geq t_0$ , the paths of  $x_1$  and  $x_2$  belong to the subset  $U$ .

Observability is not affected by the inputs variations, therefore an observable LTI system is globally observable. In the case of nonlinear systems, local observability is considered.

**Observability rank conditions :** As defined for controllability in the case of LTI systems, observability can be tested by determining the rank of the observability matrix. The observability space  $O$  is generated by the constant matrix :

$$O = \begin{pmatrix} C & CA & CA^2 & \dots & CA^{n-1} \end{pmatrix}^\top \quad (1.132)$$

the Kalman condition is then given by :

$$\text{rank}(O) = n. \quad (1.133)$$

This condition was generalised to Multiple-Input and Multiple-Output (MIMO) systems [27]. The extension to nonlinear systems was studied.

**Observability of nonlinear affine systems :** Considering the following system,

$$\begin{cases} \dot{x} &= f(x) + g(x)u \\ y &= h(x, u). \end{cases} \quad (1.134)$$

The Lie derivative expression is given by:

$$L_f h(x, u) = \sum_{i=1}^n f_i(x) \frac{\partial h}{\partial x_i} + \frac{\partial h_i}{\partial u} u. \quad (1.135)$$

The observability rank property allows to define the local observability of (1.134) if the following conditions are satisfied:

$$\text{Rank} \begin{pmatrix} L_f h_1(x, u) \\ L_f^1 h_1(x, u) \\ \vdots \\ L_f^{k_1-1} h_1(x, u) \\ \vdots \\ L_f h_p(x, u) \\ \vdots \\ L_f^{k_p-1} h_p(x, u) \end{pmatrix} = n, \quad (1.136)$$

where  $L_f^1 h = dL_f h$  is the co-vector given by the general form:

$$dL_f^j h = \left( \frac{\partial L_f^j h}{\partial x_1}, \frac{\partial L_f^j h}{\partial x_2}, \dots, \frac{\partial L_f^j h}{\partial x_n} \right), \quad (1.137)$$

$k_i$  are the output derivative levels and also called the observability index. For every system, the corresponding  $k_i$  indexes need to be compliant with the following statement [67].

**Definition 1.15. (Observability indexes)** *The naturals  $(k_1 \dots k_p)$  are the observability indexes if they are compliant with the following properties:*

- $k_1 \geq k_2 \geq \dots \geq k_p$ ,
- $\sum_{i=1}^p k_i = n$ ,
- *the Kalman criterion is satisfied.*

#### 1.3.4.2 The Luenberger Observer

The Luenberger observer is one of the most famous linear state estimators used in model-based fault detection [4, 65]. It allows reconstruction of the state variables based on a linear model of the system. Consider the following linear system:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx. \end{cases} \quad (1.138)$$

The corresponding state observer is then given by:

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \\ \hat{y} = C\hat{x}, \end{cases} \quad (1.139)$$

where  $L$  is the observer gain. The state estimation error  $e_x = x - \hat{x}$  satisfies:

$$\dot{e}_x = (A - LC)e_x. \quad (1.140)$$

If the matrix  $(A - LC)$  is stable, then the state estimation error tends to zero. This is true if  $(C, A)$  is observable. Considering a fault vector  $w_f$ , impacting the state.



System (1.138) becomes then:

$$\begin{cases} \dot{x} = Ax + Bu + Ew_f \\ y = Cx, \end{cases} \quad (1.141)$$

and equation (1.140) becomes:

$$\dot{e}_x = (A - LC)e_x + Ew_f. \quad (1.142)$$

The state estimation error has become fault-sensitive and the output estimation error  $e_y = y - \hat{y}$  can be used as a residual for fault detection.

**Remark 1.4.** *The use of observers for diagnosis does not necessarily require every state to be observable from the state vector. If a measurement exists for an observable state, a residual can be generated and fault detection can be processed for this state.*

An observer which includes all the inputs and outputs of a system is called a *Simplified Observer Scheme (SOS)*. This kind of observer does not allow fault localization, since the states are sensitive to every type of faults. Therefore several rows of observers need to be synthesised in order to be sensitive to one particular fault, which are known under the *Generalized Observer Scheme (GOS)* (Figure 1.17) and the *Dedicated Observer Scheme (DOS)* (Figure 1.16). The DOS takes only one input

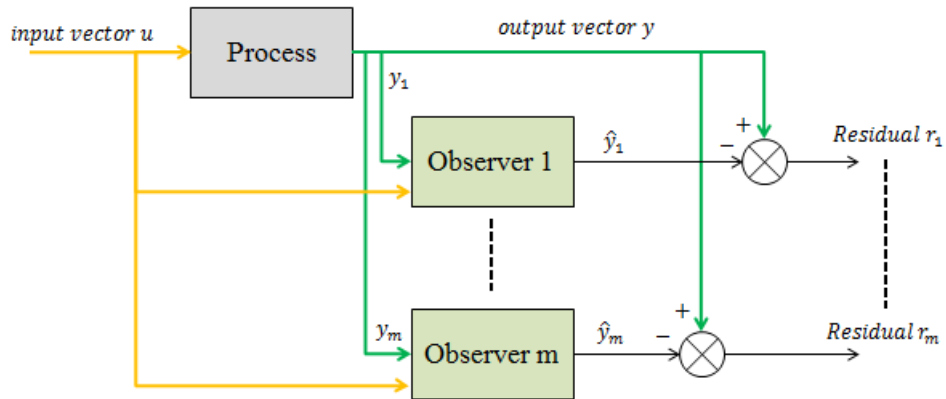


Figure 1.16: Dedicated Observer Scheme (DOS)

(or output) and is sensitive to only one specific sensor (or actuator) fault. In the

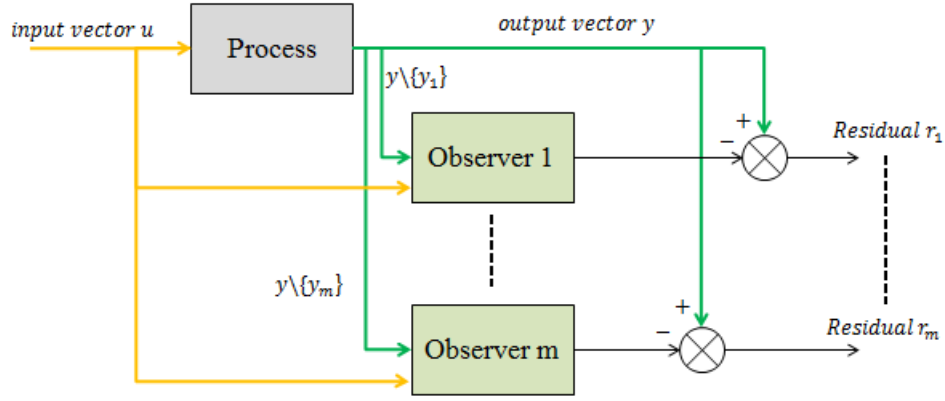


Figure 1.17: Generalised Observer Scheme (GOS)

case of the (*GOS*), all the system's inputs (or outputs), except one, are required and if a fault occurs, the observer that doesn't take into account the fault sensitive output (or input) will not be affected. This is how the fault is then detected and localised or identified.

#### 1.3.4.3 *The Standard Kalman Filter (SKF)*

The Kalman filter [32] is used for estimating states when stochastic signals are considered. This linear states estimator takes into account state and measurements disturbances and integrates a linear state-space model. The Kalman filter's correction and update steps allow generation of a predicted state vector, which permits to process the residual. This residual can then be used for the diagnosis of system faults. When discrete models are considered for System (1.141), the prediction step

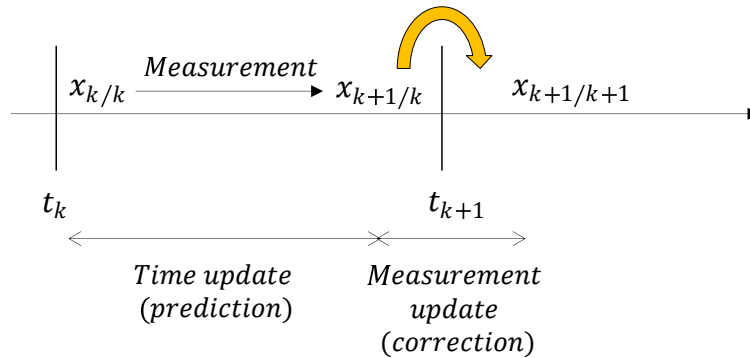


Figure 1.18: Time and measurement updates of the Kalman filter

is given by:

$$\begin{cases} \hat{x}_k^- &= A_k \hat{x}_{k-1} + B_k u_k \\ P_k^- &= A_k P_{k-1} A_k^T + Q_k, \end{cases} \quad (1.143)$$

where  $Q_k$  and  $R_k$  are respectively the state and measurement noise covariance matrix given by:

$$\begin{cases} Q_k \delta(l) &= E[w(k)w(k+l)^T] \\ R_k \delta(l) &= E[v(k)v(k+l)^T] \\ E[w(k)v(k+l)^T] &= 0, \quad (\text{with } \delta(l)=1 \text{ if } l=0; 0 \text{ otherwise}). \end{cases} \quad (1.144)$$

Here,  $w(k)$  and  $v(k)$  are the Gaussian white noises corresponding to matrices  $Q_k$  and  $R_k$  respectively. The correction step is given by equation (1.145)

$$\begin{cases} K_k &= P_k^- C_k^T (C_k P_k^- C_k^T + R_k)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k (y_k - C_k \hat{x}_k^-) \\ P_k &= (I - K_k C_k) P_k^-. \end{cases} \quad (1.145)$$

Kalman filters can be processed for synchronous motors described in [5], but also for asynchronous motors. For nonlinear systems, the Extended Kalman Filter is also a very good solution and has many applications in industry [96].

#### 1.3.4.4 Unknown Input Observers (UIO)

Another very powerful state estimator is the *UIO* because it fits well with deterministic and stochastic models. The aim of the *UIO* is to estimate the state vector while minimizing the influence of unknown inputs such as noise, which are decoupled. The structure of an *UIO* is given by :

$$\begin{cases} \dot{\hat{x}} &= F \hat{x} + T B u + (K_1 + K_2 y) \\ r_y &= (1 - C H) y - C \hat{x} \end{cases} \quad (1.146)$$

where the decoupling matrices  $F, T, K_1, K_2, H$  must be chosen in order to respect asymptotic convergence such as :

$$\begin{cases} (HC - 1)E_d &= 0 \\ T &= 1 - HC \\ F &= A - HCA - K_1C \quad \text{is stable} \\ K_2 &= FH. \end{cases} \quad (1.147)$$

Regarding nonlinear systems, the *Extended UIO* is also widely used such as the *EKF*. For strongly nonlinear systems, extensions were developed for Lipschitz-non-linearities [81, 2].

### 1.3.5 Observers for nonlinear control-affine systems

The case of nonlinear control affine systems has been studied the past 50 years [60, 40, 24] in order to design nonlinear observers.

#### 1.3.5.1 High Gain Observers

High gain observers [48, 41] are designed to minimize the nonlinearities of the corresponding system by applying a high gain on the linear terms of the system. Single output systems are considered for this example with:

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x). \end{cases} \quad (1.148)$$

System (1.148) is supposed to be uniformly observable, implying the manifold  $\psi(x)$  such as:

$$\zeta = \psi(x) = \begin{pmatrix} h(x) \\ \vdots \\ L_f^{n-1}h(x) \end{pmatrix}. \quad (1.149)$$

The nonlinear system [6] is then expressed with  $\zeta$  and  $x$  by:

$$\begin{cases} \dot{\zeta} = A\zeta + \Gamma(\zeta) + G(\zeta)u \\ y = Cx \end{cases} \quad (1.150)$$

$$\text{where, } A = \begin{pmatrix} 0 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & 0 \\ \vdots & & \ddots & 1 \\ 0 & \dots & \dots & 0 \end{pmatrix}, \Gamma(\zeta) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \gamma(\zeta) \end{pmatrix}, G(\zeta) = \begin{pmatrix} \bar{g}_1(\zeta_1) \\ \bar{g}_2(\zeta_1, \zeta_2) \\ \vdots \\ \bar{g}_n(\zeta_1, \dots, \zeta_n) \end{pmatrix}$$

$$\text{and } C = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix}.$$

It was shown by Gauthier *et al* that the following system :

$$\dot{\hat{x}} = f(\hat{x}) + \sum_{i=1}^m g_i(\hat{x})u_i - \left( \frac{\partial \psi}{\partial x}(\bar{x}) \right)^{-1} S_\theta^{-1} C^T (h(\hat{x}) - y) \quad (1.151)$$

is a high gain observer for (1.148), where  $S_\theta$  is the solution of :

$$C^T C = \theta S_\theta + A^T S_\theta + S_\theta A. \quad (1.152)$$

### 1.3.5.2 Sliding-mode observers

Sliding-mode observers [36, 100, 99, 28] are observers given by :

$$\begin{cases} \dot{\hat{x}} = f(\hat{x}, u) + \Lambda Sgn(y - \hat{y}) \\ \hat{y} = h\hat{x} \end{cases} \quad (1.153)$$

where the correcting part is not continuous, denoted by :

$$sgn(x) = \begin{cases} x & \text{if } x > 0 \\ -x & \text{if } x < 0 \\ \text{not defined for } x = 0 \end{cases} \quad (1.154)$$

$\Lambda$  is a  $n \times p$  matrix and

$$Sgn(y - \hat{y}) \triangleq \begin{pmatrix} sgn(y_1 - \hat{y}_1) \\ sgn(y_2 - \hat{y}_2) \\ \vdots \\ sgn(y_p - \hat{y}) \end{pmatrix}. \quad (1.155)$$

### 1.3.5.3 The Extended Kalman Filter (EKF)

As shown in the linear observer section, the Kalman filter is used to estimate the states of a system when disturbances occur on the measurements and the input signals if the system is observable. The Extended Kalman Filter [70, 69] is an extension of the standard Kalman filter to nonlinear systems. The non-linearities are linearised locally with tangent derivatives with Jacobian matrices. Consider the following discrete nonlinear system:

$$\begin{cases} x_{k+1} &= x_k + T_s f(x_k, u_k) + w_k \\ y_k &= h(x_k) + v_k \end{cases} \quad (1.156)$$

where:

- $\hat{x}_{k/k}$  is the state estimated at  $t_k$ ,
- $\hat{x}_{k+1/k}$  is the state estimated at  $t_{k+1}$  without correction,
- $\hat{x}_{k+1/k+1}$  is the state estimated at  $t_{k+1}$  with correction.

**Time update (prediction) :** During the time update step, the state vector is estimated at time  $(k+1)$  regarding the state and the measurements realised at time  $(k)$  such as:

$$\hat{x}_{k+1/k} = \hat{x}_{k/k} + T_s f(\hat{x}_{k/k}, u_k). \quad (1.157)$$

The prediction error covariance matrix  $P$  is also processed in this step with :

$$P_{k+1/k} = Aj_k P_{k/k} Aj_k^T + Q_k \quad (1.158)$$

with  $Aj_k$  and  $Hj_k$  the Jacobians of  $f$  and  $h$  given by :

$$Aj_k = \left. \frac{\partial (\hat{x}_{k/k} + T_s f(\hat{x}_{k/k}, u_k))}{\partial x} \right|_{x_k = \hat{x}_{k/k}} ; \quad Hj_k = \left. \frac{\partial (h(x_k))}{\partial x} \right|_{x_k = \hat{x}_{k/k}} . \quad (1.159)$$

**Measurements update (correction) :** In this step, the Klamman gain is processed where :

$$K_{k+1} = P_{k+1/k} Hj_k^T (Hj_k P_{k+1/k} Hj_k^T + R_k)^{-1}. \quad (1.160)$$

The correction of the state vector is then given by the next equation :

$$\hat{x}_{k+1/k+1} = \hat{x}_{k+1/k} + K_{k+1} (y_{k+1} - Hj_k \hat{x}_{k+1/k}). \quad (1.161)$$

The last equation of the correction step consists in updating the prediction error covariance matrix,  $P$  such as :

$$P_{k+1/k+1} = P_{k+1/k} - K_{k+1} Hj_k P_{k+1/k}. \quad (1.162)$$

### 1.3.6 Robustness and performances evaluation

In order to demonstrate the robustness of the model-based diagnosis approach, the linearisation process has to be stable at each state value (including at the system equilibrium). The equivalent linear model obtained by endogenous feedback in subsection (1.2.3.4) is not affected by any non-linearity. For this study, the diagnosis is realised with a cumulative sum and threshold comparison. If the residual exceeds the defined threshold, then an alarm is generated. The threshold levels are compared with two approaches. First, a linearisation of the stepper motor model is realised

with an endogenous feedback. The equivalent system is then used with a standard Kalman filter for fault detection by residuals generation of the flat outputs. The second approach uses an EKF with the nonlinear model of the stepper motor. The two linearisation approaches are then compared in Chapter 3.

## **1.4 Conclusion**

In this chapter, fundamentals of differential flatness were presented and an example was given with the processing of a hybrid stepper motor's flat outputs. Different linearisation approaches were given such as endogenous feedback, pseudo-linearisation and tangent linearisation. The problem of local linearisation around an equilibrium point is solved for certain nonlinear systems with cancellation of their non-linearities by diffeomorphism and endogenous feedback. An example of linearisation of a stepper motor was shown in the first section. In the second section, model-based diagnosis tools such as residual generators, state estimators and observers for linear and nonlinear systems were shown.

In the next chapter, an aeronautical case study is described. The case of an electromechanical actuator used in a fuel circuit of a turboshaft engine will be presented. A new actuator monitoring architecture then suggested. The safety assessment of this new architecture will be discussed.



# Chapter 2

## Proposal of a flight-critical monitoring system

### Contents

---

<b>2.1</b>	<b>Introduction . . . . .</b>	<b>65</b>
<b>2.2</b>	<b>Flight critical systems overview . . . . .</b>	<b>66</b>
2.2.1	Definitions . . . . .	66
2.2.2	Aviation safety standards and airworthiness . . . . .	72
2.2.3	System engineering methodologies . . . . .	74
2.2.4	Leading system engineering tools . . . . .	77
<b>2.3</b>	<b>Safety assessment of a turboshaft fuel valve actuator . . . . .</b>	<b>80</b>
2.3.1	Current system description . . . . .	80
2.3.2	Failure Modes and Effects Analysis . . . . .	82
2.3.3	Fault Tree Analysis (FTA) . . . . .	85
2.3.4	Failure rate processing . . . . .	88
2.3.5	Development Assurance Level assessment . . . . .	88
2.3.6	Current actuator redundancies in flight control . . . . .	90
<b>2.4</b>	<b>Proposal of a safe and robust architecture . . . . .</b>	<b>94</b>
<b>2.5</b>	<b>Conclusion . . . . .</b>	<b>97</b>

---

---

**Abstract:** In Chapter 1, it was recalled that monitoring systems designs of flight control systems are in certain cases based on nonlinear analytical diagnosis methods. To reduce complexity and processing costs, different linearisation methodologies (pseudo and exact linearisation) were presented. Among them, linearisation by endogenous feedback of flat systems allow to determine an equivalent linear and stable system.

In this chapter, flight-critical systems are introduced. It is shown that analytical monitoring allows to reduce the number of redundancies according to specific guidelines of aeronautical safety standards such as ARP-4754. This innovation has been applied to a flight-critical actuator of a turbo-shaft engine.

---

## 2.1 Introduction

The rapid growth in the volume of air traffic over the past decades, coupled with the ever-present mandatory objective to reduce the number of fatal aircraft accidents has led to a significant increase in aircraft safety requirements and regulations. In addition, compliance with the safety regulations must be demonstrated through a complex certification process, and significantly impacting system engineering methodologies for modern avionic equipment. The design and development of Safety Critical Systems (SCS), such as avionic equipment, Engine Computer Units (ECU), Flight Computer Units (FCU), actuators, direct-drives, servo-valves or electronic components (Figure 2.1) is today challenged by the severe environmental constraints, high performance standards and rugged safety requirements on software, hardware and system architectures [91, 92, 94]. Since any failure occurring on a safety-critical system can lead an aircraft to a catastrophic event (failure causes a crash) due to non-availability and/or non-integrity causes, FCS equipment must be designed to fully comply with all safety requirements starting from the inception phase.

The aim of this chapter is to describe the realisation of a possible flight-critical actuator architecture based on a combination between analytic and material redundancy. Terms and definitions related to flight critical systems are first recalled in section 2.2. Leading FCS design methodologies and tools are also presented in order to explain how safety standards are taken into account in the design of complex FCS. A case study is realised in section 2.3 on a fuel valve actuator of a biturbine helicopter. The actuator safety analysis shows the impact of a possible fault occurrence on the biturbine which explains the criticality levels of such actuators. Finally, a new safety compliant actuator monitoring architecture is proposed in section 2.4. This architecture has also been subject to a patent [13] developed with *THALES Systèmes Aéroportés* and the *IMS* and *ESTIA-Recherche* Laboratories.

## 2.2 Flight critical systems overview

As an introduction to this study, relevant safety related definitions regarding FCS are recalled in this section. These terms are defined by SAE standards [91, 92] regarding environment, hardware and software considerations. Other terms are also described in [68, 16].

### 2.2.1 Definitions

**Safety and flight-critical systems :** [16] In the event of their failure, these are the systems that may either directly or indirectly lead to situations in which human life is put at risk, damage to the natural environment occurs, or large economical loss is suffered. Complex safety or flight-critical systems are defined as those that cannot be shown only by test or where the logic is difficult to comprehend without the aid of analytical tools.

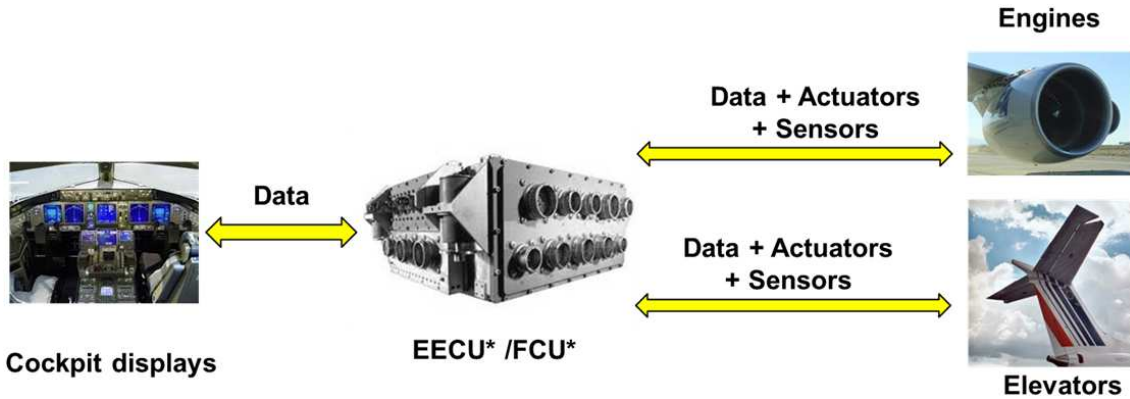


Figure 2.1: Some safety critical systems in current aircraft

**Complexity :** System complexity grows together with the size of the software applications, the number of functions, and the number of states of the system, often making comprehension and verification of digital functions difficult. Also, discrete behaviour is related to complexity of highly integrated systems because small input variations may cause greater variations on the system's output. Verification and validation of such complex functions become therefore more difficult. Another term which specifies complexity is invisibility. Indeed, as software is not physically visi-

ble, it has to be represented by many different Unified Modelling Language (UML) diagrams and visualised by overlapping different functional views such as data flow or control flow. Complex and high integrity systems were developed in many industrial fields such as aerospace, automotive and many transport domains. An example of complexity growth is shown by the increase of flight software size (number of Non-Comment Source Lines (NCSL)) in NASA space missions (Figure 2.2) :

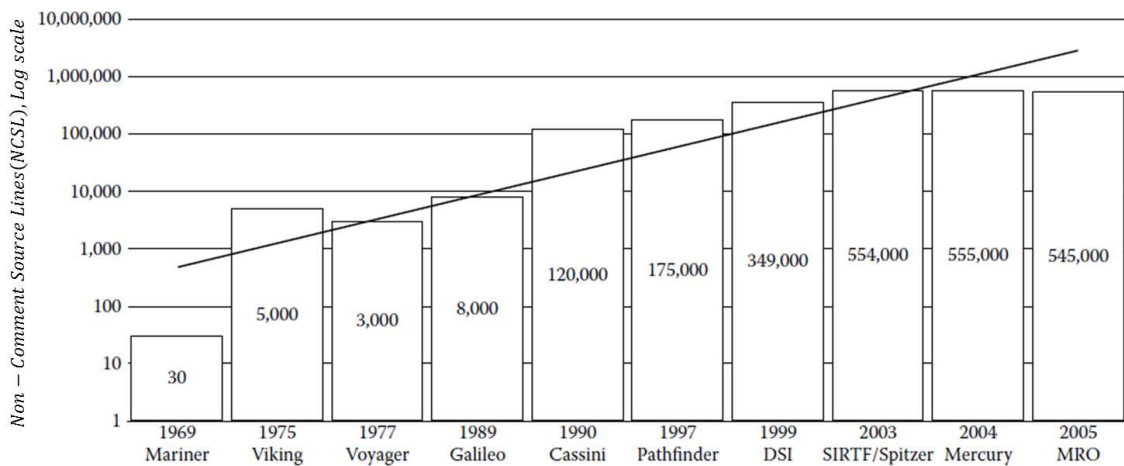


Figure 2.2: Complexity-related software growth in NASA space missions, (Source: From Dvorak, D.L., Editor (2009). NASA Study on Flight Software Complexity)

**Availability :** The availability of a system can be evaluated by the probability of its working correctly over a certain time frame.

**Security :** A secure system does not permit the occurrence of unauthorized access to information. Regarding security, the feared occurrence is a malicious attack but regarding safety the feared event is a failure.

**Integrity :** Integrity issues appear if there is an occurrence of inappropriate information alterations. As an example, data integrity refers to the possibility that a system will detect faults and recover by correcting the resulted errors. If a computer unit processes erroneous data, the system's integrity is not satisfied.

**Reliability :** Reliability is the ability for a given system to operate correctly over a given period of time. Reliability depends on time and can also be defined as a failure rate, where the failure rate corresponds at the time when the systems encounters a failure. For example, safe systems are not necessarily reliable. If a system is fail-

safe, there is no failure occurrence possible, therefore the reliability is null. In a series of system blocks, the reliability function  $R$  is given by :

$$R(t) = \prod_{i=1}^N r_i(t). \quad (2.1)$$

For parallel blocks, reliability becomes:

$$R(t) = 1 - \prod_{i=1}^N (1 - r_i(t)), \quad (2.2)$$

where  $r_i$  is the number of failures in the  $i^{th}$  data group or subsystem block. Many

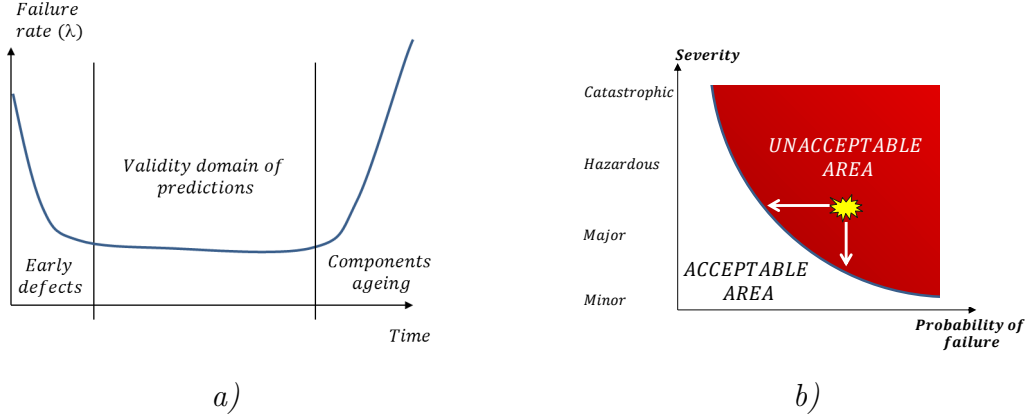


Figure 2.3: a) Failure rate variations; b) Aviation risk acceptability

components vary as shown in figure 2.3-a. Their lifetime is divided in three domains. First, components show high failure rates related to possible defects that remain from the manufacturing phase, also called *infant mortality*. The end phase is called the *wear-out* and is related to ageing. The failure occurrence probability  $P$  follows an exponential law given by:

$$P(t) = 1 - e^{-\lambda t} \quad (2.3)$$

where  $\lambda$  is the failure rate (the amount of failures during operating time). If  $\lambda t \leq 10^{-2}$  then  $P(t) \approx \lambda t$ . For a given system, failure rates and the severity levels are related and allow to define an acceptability area (Figure 2.3-b). Let's assume that a failure occurrence leads the system into the unacceptable domain. Generally, it is only possible to get back to the acceptable area by reducing the failure rate

which implies to reduce the severity level.

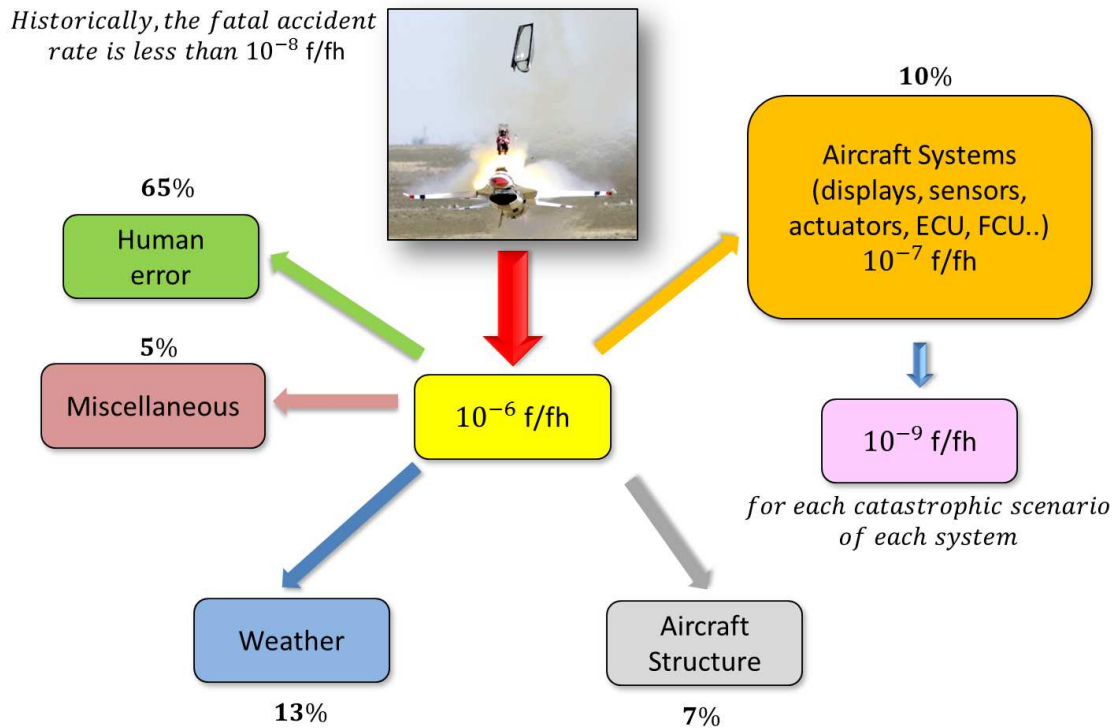


Figure 2.4: Main aviation fatal accident causes

**Incident and accident :** An incident is an undesired event which might lead to integrity issues such as erroneous data flow for example. The catastrophic accident of the NASA space shuttle Challenger was the result of chain events implying the incident of lift-off when the ambient temperature of the fuel was low. As a result, the flight 51-L exploded 73 seconds after lift-off causing human losses and the loss of a multi-billion-dollar shuttle. As shown in figure 2.4, the origin of aircraft accidents is related to multiple causes. The main accident cause is due to human error. Flight critical systems such as cockpit displays, Flight Computer Unit (FCU) or Electronic Engine Computer Unit (EECU), have to be affected by a very low failure rate per flight hour.

**Dependability :** [16] A dependable system can be defined as a system for which reliance can justifiably be placed on the service it delivers. Dependability is also defined by its attributes which are safety, reliability, availability and security.

**Airworthiness :** This term is used to regroup aircraft related regulations (FAA and EASA), standards, safety design rules and certification processes.

**Fail-Safe :** Fail-safe systems are systems able to recover in a fail-safe state after a single or multiple failure occurrence.

**Fault avoidance :** In the system design phase, formal methodologies are used to ensure that a system is fault free.

**Fault removal:** Aims to remove faults from a system once they have been entered as a consequence of improper design or wrong implementation.

**Fault detection:** In this case, faults are detected while the system is active.

**Fault prediction :** Fault prediction consists in evaluating the likelihood that a given system will fail at a certain time. Fault prediction can be used for evaluating the probability of a failure occurrence or to realise a system verification test after an estimated time period (Figure 2.5).

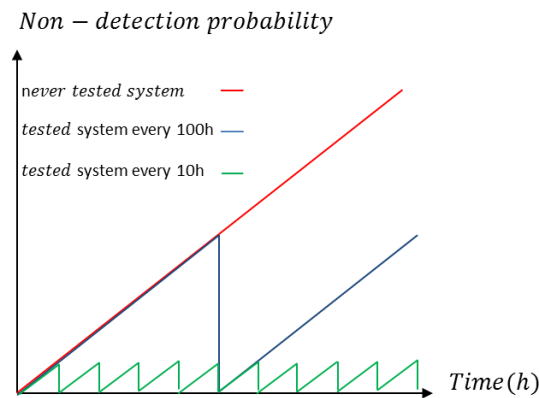


Figure 2.5: Probabilities of failure occurrence with and without test

**Fault tolerance :** It qualifies a system which is able to operate correctly in the presence of faults. Redundant systems are often fault tolerant.

**Fault coverage :** It is related to fault -detection, -avoidance,-removal,-tolerance or prediction and is a measure of the degree of success of each of these functions.

**Active failures :** It is a failure which is detected during the system's activity.

**Hidden, dormant or latent failures :** For flight critical systems, dormant failures are not detected during the flight. As an example, the loss of monitoring including passivation means or the loss of redundant paths are considered as dormant failures.

**Human error :** Systematic faults generated by human activity (i.e. a mistake in specifications, design, manufacturing, a mistake in operating or during maintenance



actions).

**System error** : It is the consequence of a fault occurrence (Figure 2.6). As an example, a fault caused an inability for a system to open a valve on command, which is considered as an error.

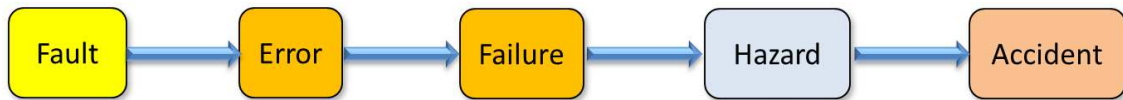


Figure 2.6: Possible fault consequences

**Dissimilarity** : Dissimilarity can be used as well in hardware design as for software. Software dissimilarity is achieved by realising two different solutions by two different teams. The two software run separately and their outputs are compared or added in order to ensure integrity.

**Development Assurance level (DAL)** : In the safety assessment process, the probability of a failure occurrence regarding the severity is allocated to grades denoted as development assurance levels. Figure 2.7 shows a severity allocation table comparing multiple variables such as the failure rate and the probability range.

Failure rate per hour					
Probability range	FAA & EASA	Probable		Improbable	
		Frequent	Reasonably probable	Remote	Extremely remote
Severity	FAA & EASA	MINOR		MAJOR	SEVERE MAJOR HAZARDOUS
Failure condition effect	FAA & EASA	<ul style="list-style-type: none"> <li>- Slight reduction in safety margins</li> <li>- Slight increase of crew workload</li> <li>- Some inconvenience to occupants</li> </ul>		<ul style="list-style-type: none"> <li>- Significant reduction in safety margin</li> <li>- Significant increase in crew workload</li> <li>- Some discomfort to occupants</li> </ul>	<ul style="list-style-type: none"> <li>- Large reduction in safety margin</li> <li>- Higher workload or physical distress for the crew</li> <li>- Serious or fatal injury for occupants</li> </ul>
DAL	ARP 4754 DO178 DO254	D		C	B
					A

Figure 2.7: FAA and EASA severity allocation



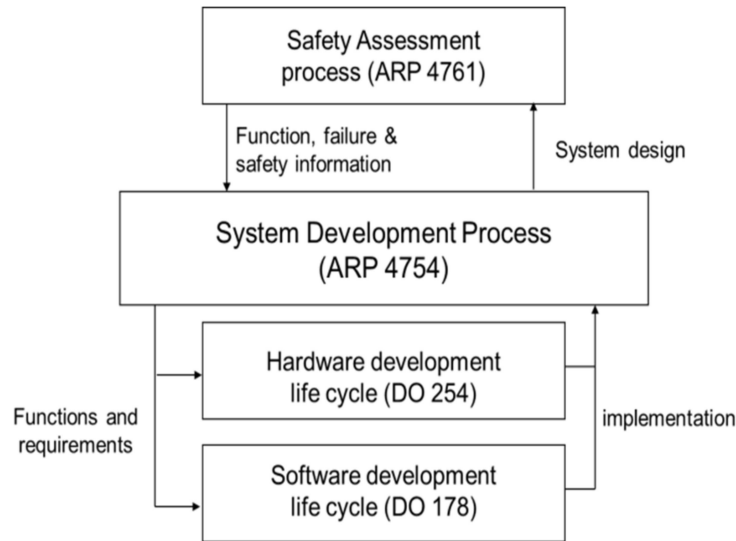


Figure 2.9: Safety integration in flight systems design

certification between others. Figure 2.9 shows how different standards are used for safe design process. In order to assess safety requirements in system design, several layers must be completed sequentially and on parallel (Figure 2.10). Routinely, a

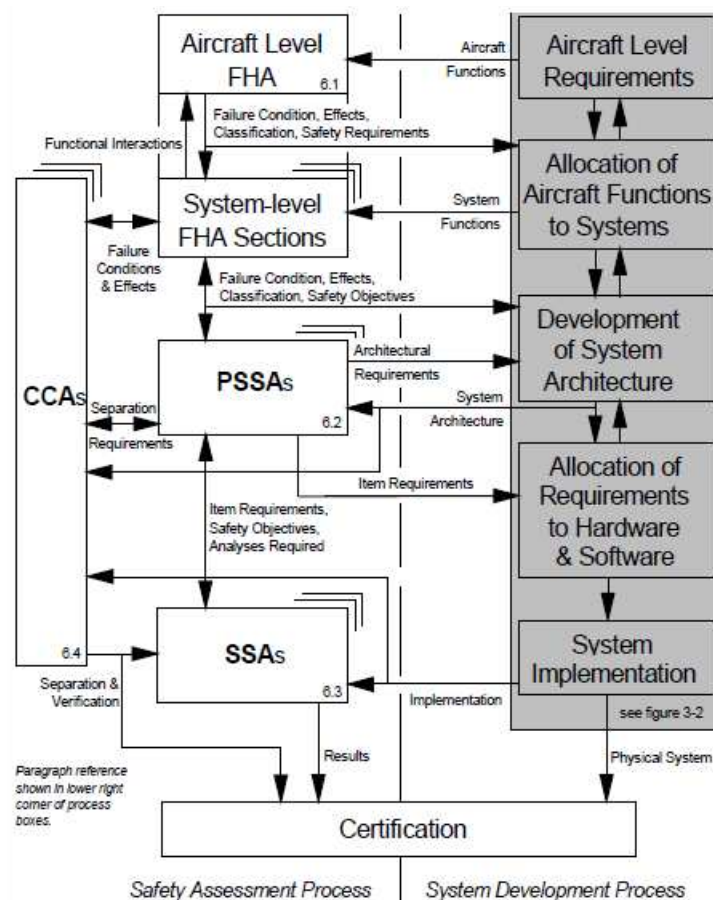


Figure 2.10: Safety assessment and development process

first reliability analysis is realized where the failure rates are processed. These rates are associated in the Failure Modes and Effects Analysis (FMEA) phase. The Preliminary Safety System Assessment (PSSA) and Safety System Assessment (SSA) phases are described in the guidelines of the SAE ARP4761 [95] standard together with the Common Causes Analysis (CCA). These guidelines help in the safety development process of the system. When these elements are established, the Fault Tree Analysis (FTA) is realised with having previously completed the Functional Hazard Assessment (FHA). The next step consist in performing the Common Mode Analysis (CMA). If multiple redundancies are required, common modes are forbidden, in order to be compliant with integrity and availability criterion. Many tools where developed in industry to perform these tasks, which are recalled in the next subsection. They are based on the knowledge of past accidents, and improvements made on previous technologies.

Summarising, aviation standards and guidelines are established by international regulation authorities (ICAO, FAA and EASA) in order to give mandatory objectives regarding environment [91], software [92], hardware [93], design [94] and safety [95] constraints. Systems integrating these constrains in their design are denoted as high integrity systems [68], implying the use of system engineering (SE) methodologies in industry.

The next section is dedicated to recall main SE methodologies including methods, tools and processes used in aerospace industry. The patented [13] architecture proposed in section 2.4 emerged from the next presented SE methodologies.

### **2.2.3 System engineering methodologies**

During the 60's, system engineering (SE) approaches have been set up in order to manage the complexity of great industrial projects (e.g. architecting NASA space transportation systems, flight control systems or engine control systems). A complexity not only defined by technological performances of systems (and software) but also including tasks definition, processes and methods, tools, need analysis,

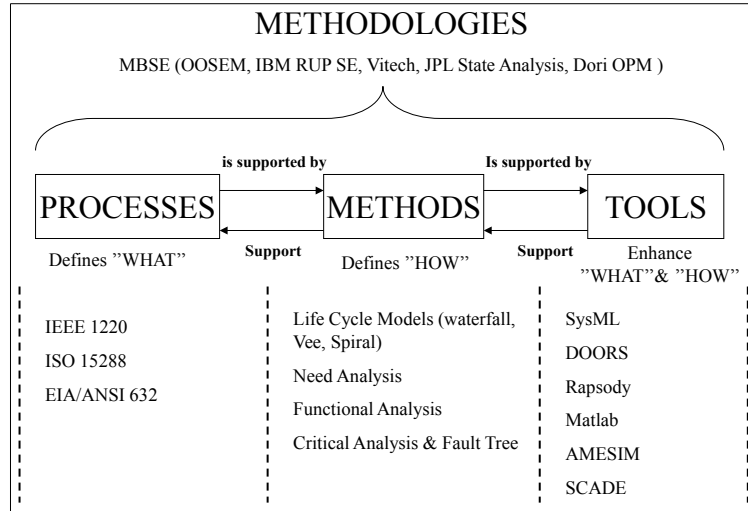


Figure 2.11: System engineering methodologies scheme

product life-cycle management, system verification and validation models, system safety requirements, collaborative work management, human skills and knowledge. According to the *Association Française de l'Ingénierie Système* (AFIS) standard, System engineering is defined as the relationship between processes, methods and tools (Figure 2.11) [31] where:

- **Processes** (Figure 2.12), are logical sequences of tasks answering to «*what task is realised?*» (e.g. conceptualizing, develop, operate and maintain, replace or dismantle), and must be compliant with the ANSI/EIA 632, ISO/IEC 15288 and IEEE 1220 standards where ISO/IEC 15288 [61] is the standard for the description of life-cycle of systems, ANSI/EIA 632 [1] is a set of processes for engineering or re-engineering a System and IEEE 1220 [56] is the standard for system management,

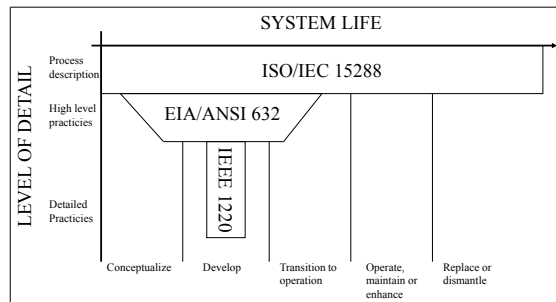


Figure 2.12: Leading process standards for system management

- **Methods** and support processes are also supported by tools; they enable the realization of systems by integrating tasks of processes, following one or several models (Royce's waterfall [90], Forsberg and Moog's Vee [44, 43], Bohem's spiral [15]). Methods answers to «*How shall tasks be done?*»,
- **Tools**, enable tasks to be processed according to a particular method. Most System Engineering Tools (SET) are software designed to assist engineers in the modelling and simulation stages of the product, known as «*Concept Stage*», «*Development Stage*», «*Production Stage*», «*Utilization and Support phases*» and the «*retirement phase*». Tools enable to answer to previous «*what?*», and, «*how?*», and enhance tasks efficiency.

According to [31], these terms are often erroneously considered with methodologies, which should be understood as a «collection of related processes, methods and tools».

### 2.2.3.1 Leading System Engineering Methodologies

SE Methodologies cover up a large scale of system conception and management methods, tools and processes in order to solve the problem of system complexity regarding the size of industrial projects and their heterogeneity, such as multi-physical domains, project costs or safety constraints. SE Methodologies also involve system integration, which implies a multitude of sub-systems constituting a global system working homogeneously within its environment. While system complexity is growing, major SE leaders such as the *International Conference on System Engineering* (INCOSE) and AFIS Societies become a reference in their discipline. Currently, Model-Based System Engineering (MBSE) methodologies depicted in table 2.1 are used in industry. Model Driven Architecture (MDA) has been adopted by the Object Management Group (OMG) to designate MBSE and relies on a shifted code-centric to model-centric development approach systems [21]. MDA's goals are basically to enable system portability, interoperability and re-usability through its architectural design approach. MBSE also enable to elevate the engineering process to the association of design, specification, integration, validation and operation of systems [31].

MBSE Methodologies	Description
<i>IBM Technologic Harmony - SE</i>	Subset of a software/system development process
<i>INCOSE Object Oriented Systems Engineering Method (OOSEM)</i>	Top-down approach, using OMG SysML used for specification, system analysis, design and verification
<i>IBM Rational Unified Process for Systems Engineering (RUP SE) for Model-Driven Systems Development (MDSD)</i>	RUP is a methodology which is both a process framework and process product from IBM Rational designed for software development projects management
<i>Vitech MBSE Methodology</i>	Methodology based on 4 interdependent activities (Source requirements analysis,functional/behaviour analysis, architecture), linked through a common system design repository
<i>JPL State Analysis (SA)</i>	Methodology emphasizing a model and state based control architecture
<i>Dori Object Process Methodology (OPM)</i>	OPM is defined as a system development approach and a life cycle support, based on Object-Process Diagrams (OPD) and Object-Processed Language (OPL)

Table 2.1: MBSE methodologies used in Industry

### 2.2.4 Leading system engineering tools

The achievement of such methodologies could not arise without dedicated tools and software. We have seen that tools support methods and that tools are part of methodologies. In many industrial domains, tools are used for various applications such as:

- Analysis and environment identification,
- Requirement Management,
- Functional & Physical Architecture,
- Component Design,
- System Performances Estimation,
- Prototyping,

- HIL (Hardware In the Loop) Simulation.

Category	SE Tools	Developer
Requirement Management	Doors	Telelogic
	Reqtify	TNI
Modelling and specification	Artisan studio	Atego
	SCADE	ESTEREL Technologies
	Rhapsody	IBM
	SysML	OMG
	Atelier B	ClearSy
Simulation & Verification	Matlab-Simulink	Mathworks
	AMESIM	LMS
	Scilab	Scilab
	Rational StateMate	IBM
	Dymola/Modelica	Dassault Systèmes
Architecture/Cosimulation	Cosimate	ChiasTek
Prototyping and HIL Simulation	DSPACE	DuraSpace
	SolidWorks,CATIA	Dassault Systèmes
	Nastran FEMAP	Sigmeo
	Syndex	INRIA
	Real-time target machine	Speedgoat

Table 2.2: Main MBSE methodologies used in industry

For each stage of the system development cycle, several tools exist (Table 2.2) and offer the possibility of having a relatively appropriate overview of the future system's functionalities, performances regarding cost, safety and reliability. The following example (Figure 2.13) [66] points out the integration of some of the previous tools in the SE process.

The OMG SysML<sup>TM</sup> (System Modelling Language) tool, which is one of the most used tool in industry, is a Model-Based integration platform that performs



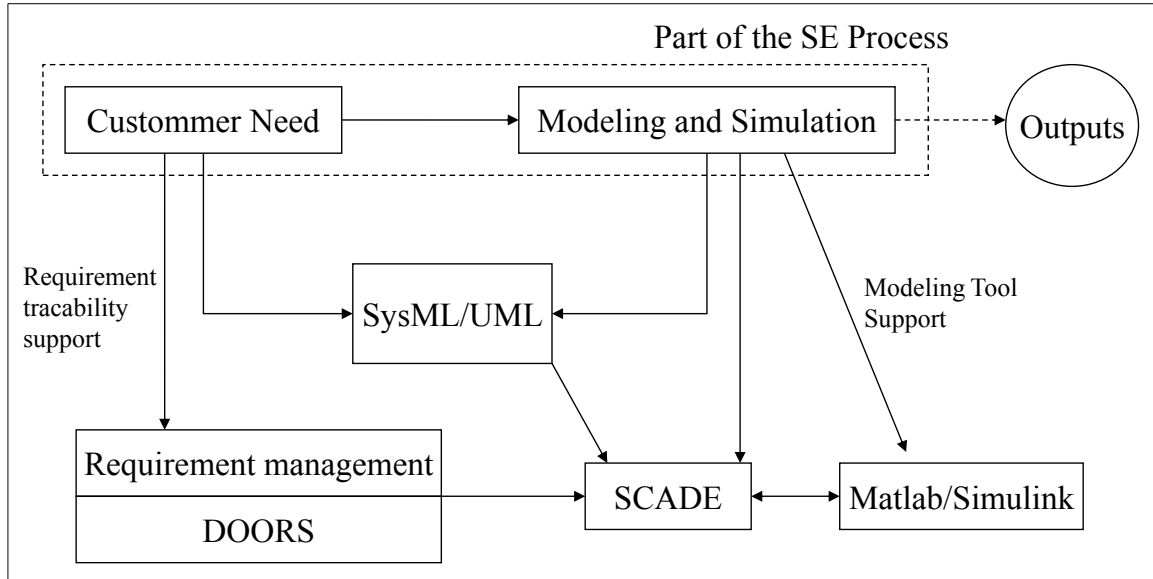


Figure 2.13: Tool integration in the SE process

the integration of Dynamical, Cost, Manufacturing and CAD (Computer Aided Design) models. Integration is enabled through a Framework, which is made of the main system development phases, such as System Requirements, Functional Analysis, Simulations & Engineering and System Architecting. SE approaches used in industry consist in elaborating a functional need analysis followed by a functional architecture, which is then submitted, to experts who are affected to one specific non-functional viewpoint (e.g. availability, fault tolerance, system integrity).

The use of SE methodologies is a valuable asset in the design of complex and high-integrity systems. It was shown in this section that leading SE tools and methods allow to realise multilayer aircraft systems designs. These systems must also integrate requirements and constraints defined by aviation regulation standards. The respect of safety requirements in the design of flight-critical systems is a major task in aerospace industry and is realised with the presented SE tools and methods.

The next section presents the application of safety assessment of a flight-critical actuator used in a helicopter engine based on a SE method. The severity is evaluated regarding aviation regulation standard ARP4761 [95].

## 2.3 Safety assessment of a turboshaft fuel valve actuator

The aim of this analysis is to confirm that a fault occurrence on the fuel system actuator might lead to a catastrophic event, characterizing therefore the actuator as flight-critical. Failure modes and rates of the actuator will be determined in this study regarding the NSWC [88] standard. The fault tree analysis (FTA) of the fuel system permits then to process the failure rate at each subsystem level. This section also shows the required redundancy level of a safe fuel system actuator.

### 2.3.1 Current system description

In order to meter the fuel flow in helicopter engines, also called turbo shaft engine, different components and sub-systems interact in the global metering system architecture (Figure 2.14):

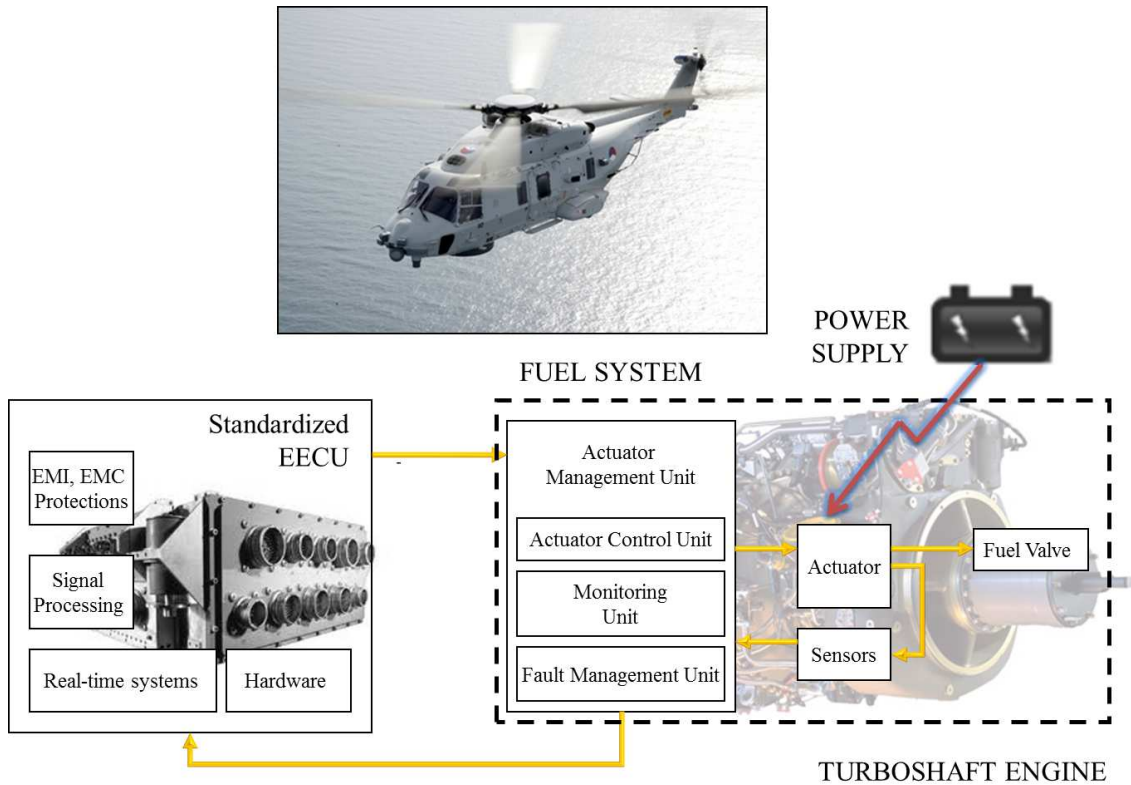


Figure 2.14: Turboshaft fuel control scheme

- **The Electronic Engine Control Unit (EECU)** is an electronic device on board of the helicopter that manages engine related sensors and actuators and integrates the fuel actuator control unit, monitoring functions and fault management control;
- **The actuator**, an electromechanic or hydromechanic drive, is enslaved by the EECU. It receives the rotation command related to the fuel flow valve. The motion and position of the fuel valve is given by the actuator in order to change the fuel flow (Figure 2.15);

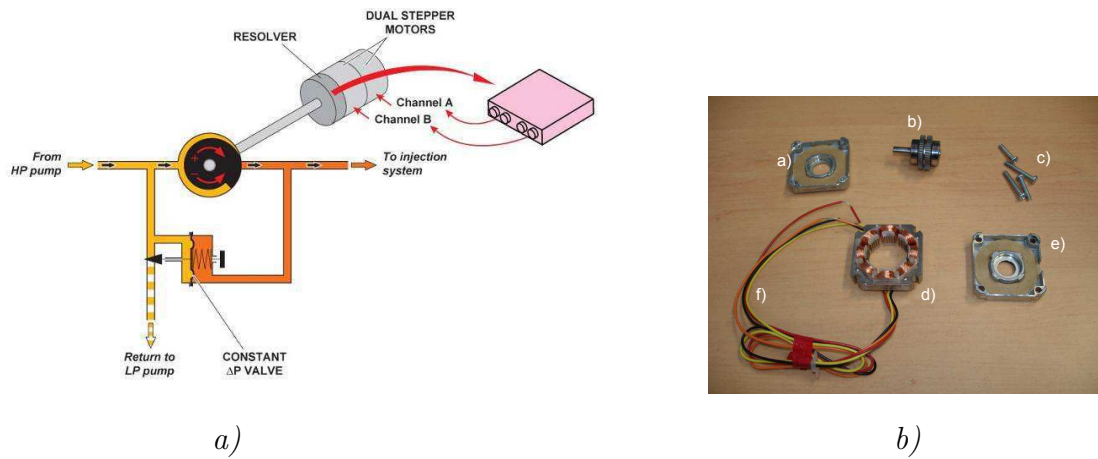


Figure 2.15: a) Fuel metering system(source: Turboméca) ; b) Stepper motor components

This fuel valve actuator is made of a hybrid stepper motor containing a hull (a) and (e), a rotative shaft (the rotor) mounted with its bearings (b) and made of two magnetized gears. The stator (d) is made of windings (f) in order to generate electromagnets when the wires are electrically supplied.

- **Sensors** (e.g. Hall-Effect Sensors) are used for phase commutation with brushless motors and also position monitoring.

This architecture reveals different types of constraints due to its multi-physical nature and the multidisciplinary related to each block unit, regarding:

- *Safety*, which implies system availability and integrity;

- *Equipment and software certification*: the DO-178B [92] standard implies software equipment certification, while the electronic and mechanic components have to be validated by testing them directly (heat, pressure, EMC, EMI);
- *System performances*, consisting of various kinds: temporal, where the aim is to ensure the stability, precision and response time (e.g. actuators are time-critical airborne systems therefore their response time must be less than 10ms). Also the frequency domain is concerned regarding signal bandwidth, noise and disturbances. Other important performance criteria relies on the mechanical part regarding mass and energy optimization;
- *Fault tolerant control*, fault diagnosis, fault detection and isolation due to many causes (mechanical, electrical, vibrations). For avionics, failure rates are about  $10^{-9}$  to  $10^{-5}$  per flight hour depending on the equipment criticality level;

Each of the presented components is likely subject to failures. In order to integrate and assess failures probability in the system design, failure modes and analysis (FMEA) are realised. The next section presents the FMEA of the fuel system of a helicopter engine.

### 2.3.2 Failure Modes and Effects Analysis

The hybrid stepper motor (HSM) can be affected by multiple failures which might be related to electrical or mechanical components. The presented failure (Table 2.3) modes be based on an AC-motor, although it will be general enough to be applied to most electric motors. Therefore, regarding the stepper motor which is built in the actuator, the proposed failure modes of the NSWC standard is not to be considered for certain cases. As an example the stepper motor does not have any brushes and collector so a failure mode related to this component is not considered. Failure modes of electric motors and shown in table 2.3 are given by the Naval Surface Warfare Center (NSWC) standard [88]. Knowing the stepper motor FMEA, the Functional Hazard Assessment (FHA) starting from the stepper motor to the turboshaft engine

will be presented in figure 2.16.

FAILURE MODE	FAILURE CAUSE	FAILURE EFFECT
<ul style="list-style-type: none"> <li>- Worn bearing</li> <li>- spalling</li> <li>- creeping or spin</li> </ul>	<ul style="list-style-type: none"> <li>- Poor lubrication</li> <li>- Contamination</li> <li>- Overloading or high temperature</li> </ul>	<ul style="list-style-type: none"> <li>- Noisy</li> <li>- Heat build-up</li> <li>- Armature rubbing stator</li> <li>- Seized</li> </ul>
<ul style="list-style-type: none"> <li>- Open winding</li> <li>- Shorted winding</li> </ul>	<ul style="list-style-type: none"> <li>- Excessively high temperature</li> </ul>	<ul style="list-style-type: none"> <li>- Motor is not running</li> </ul>
<ul style="list-style-type: none"> <li>- Cracked housing</li> </ul>	<ul style="list-style-type: none"> <li>- Fatigue</li> <li>- External shock</li> <li>- Vibration</li> </ul>	<ul style="list-style-type: none"> <li>- Leakage of dust into motor</li> <li>- Shorted or seized</li> </ul>
<ul style="list-style-type: none"> <li>- Sheared armature shaft</li> <li>- Cracked rotor</li> <li>- laminations</li> </ul>	<ul style="list-style-type: none"> <li>- Fatigue</li> <li>- Misalignment</li> <li>- Bearing failure</li> </ul>	<ul style="list-style-type: none"> <li>- Seized</li> <li>- Armature rubbing stator</li> </ul>
<ul style="list-style-type: none"> <li>- Worn sleeve bearing</li> </ul>	<ul style="list-style-type: none"> <li>- Excessive load (belt tension)</li> <li>- Frequent starts and stops under heavy loads</li> <li>- Poor lubrication</li> </ul>	<ul style="list-style-type: none"> <li>- Seized</li> <li>- Noisy</li> <li>- Heat build-up</li> <li>- Armature rubbing stator</li> </ul>

Table 2.3: Electric motor failure modes

LRU	Failure Mode	Failure rate (f/fh) for one element	Effects at LRU level	Requested design evolution
HSM	Shorted/opened wires	6E-06	motor not turning	Redundancy/Monitoring
HSM	Worn bearings	1E-06	Shaft jammed Misalignment of Shaft/ oscillations	Redundancy/Monitoring
HSM	cracked housing	1E-09	Shaft jammed	Redundancy/Monitoring
HSM	broken shaft	2E-06	Shaft disconnected	Redundancy/Monitoring
Actuator	Shaft jammed	3E-06	Shaft jammed	Redundancy/Monitoring
Actuator	Misalignment of shaft/ oscillations	3E-06	Misalignment of shaft/ oscillations	Redundancy/Monitoring
Actuator	shaft not turning	6E-06	shaft not turning	Redundancy/Monitoring
Fuel valve	input command and real positon are different and out of tolerance	3E-06	Undetected drift of position	Redundancy/Monitoring
Fuel valve	valve in high flow position	9E-06	Undetected overflow position	Redundancy/Monitoring
Fuel valve	erroneous valve position	1E-05	Undetected position error	Redundancy/Monitoring
Fuel valve	valve jammed	3E-06	detected loss of valve control	Redundancy/Monitoring
Fuel circuit	lowered fuel flow (2 circuits)	9E-12	pumping on 2 engines	Redundancy/Monitoring
Fuel circuit	lowered fuel flow (1 circuit)	3E-06	pumping on 1 engine	Redundancy/Monitoring
1 engine	Lowered fuel flow and valve position error	2E-05	uncommanded IFSD	None
2 engines	Lowered fuel flow and valve position error on 2 sides	2E-10	double uncommanded IFSD	None
1 engine	pumping	3E-06	commanded IFSD	None
2 engines	pumping on the two sides	9E-12	double commanded IFSD	None
1 engine	pumping and LOPC request from the pilot for 1 engine	3E-06	LOPC requested by pilot according to situation or not	None
2 engines	pumping on 2 engines and LOPC request from the pilot for 1 engine	9E-12	Double LOPC requested by pilot according to situation or not	None
1 engine	non detected overflow in one fuel circuit and lowered fuel flow	9E-06	OSP UAC on one engine	None
2 engines	non detected overflow in one fuel circuit and lowered fuel flow on the 2 engines	8E-11	OSP UAC on the two engines	None

Figure 2.16: Failure rate processing and engine effect analysis

In order to determine the failure rate at each level of the system, the fault propagation is realised by a tree analysis starting from a single component of the HSM. The next section shows how the fault propagation affects the helicopter engine<sup>1</sup>. A second FTA related to the actuator is now detailed. The failure effects of the fuel system FTA become the final events of the stepper motor FTA (Figure 2.18).

### 2.3.3 Fault Tree Analysis (FTA)

In this section, the FTA of the fuel system and the actuator based on the FMEA was realised within the SYRENA (Turboméca) project which yields the following Fault Tree Analysis.

In the FTA, each event has a failure rate which is processed at each AND and OR gates with elementary probabilities given by the following equations :

$$\begin{aligned} P(A \cap B) &= P(A|B).P(B) \\ P(A \cup B) &= P(A) + P(B) - P(A \cap B). \end{aligned} \tag{2.4}$$

$P(A \cap B) = P(A).P(B)$  if the events are independent ( $P(A|B) = P(A)$ ),  
 $P(A \cup B) = P(A) + P(B)$  if the events are mutually exclusive ( $P(A \cap B) = 0$ ).

---

<sup>1</sup>The acronyms of the turboshaft effects are given by:  
**IFSD**: In Flight Shut-Down  
**LOPC**: Loss Of Power Control  
**OSP-UAC**: Spurious activation of two engine overspeed protections.  
**LRU**: Line Replaceable Unit

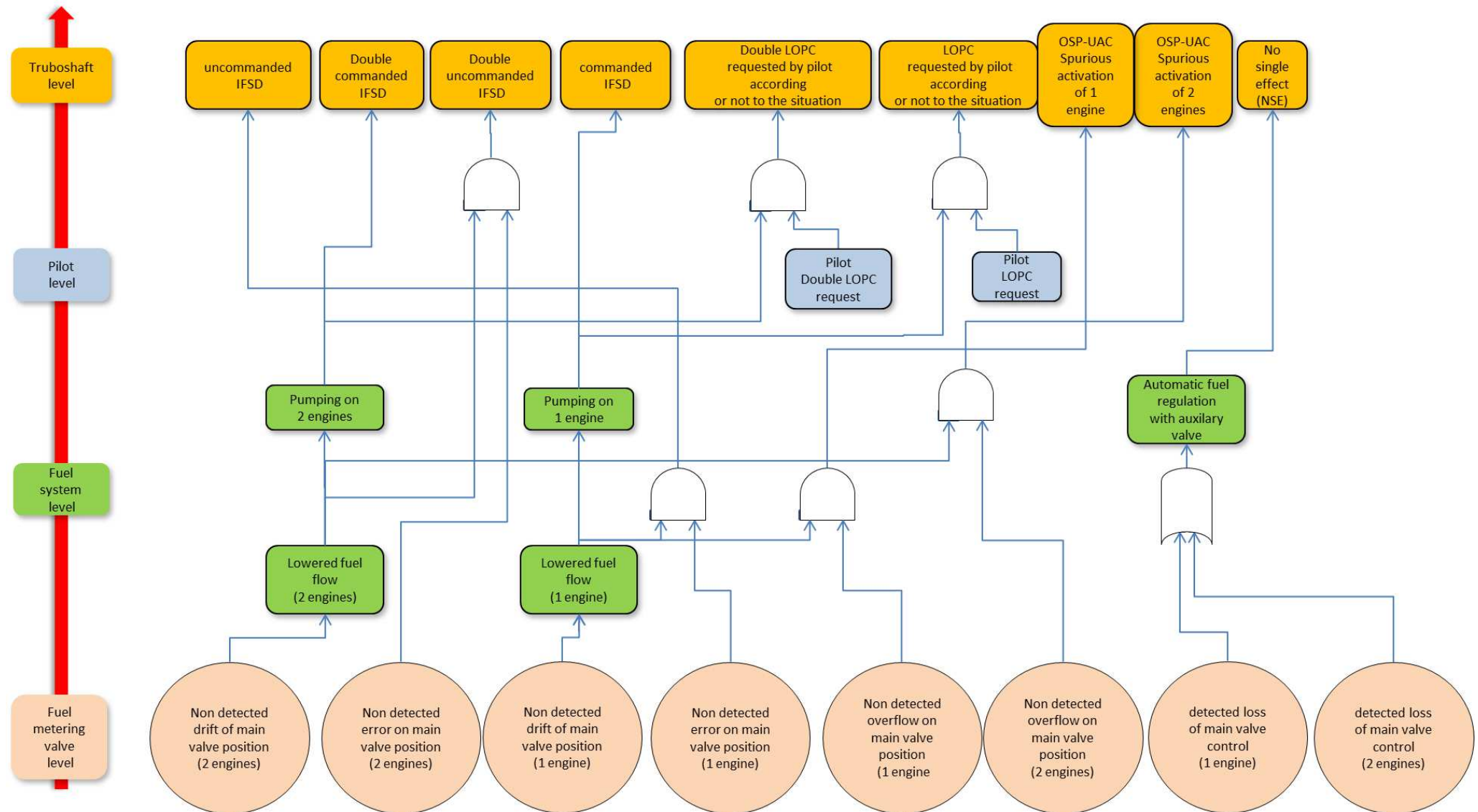


Figure 2.17: Fault tree analysis of the fuel system



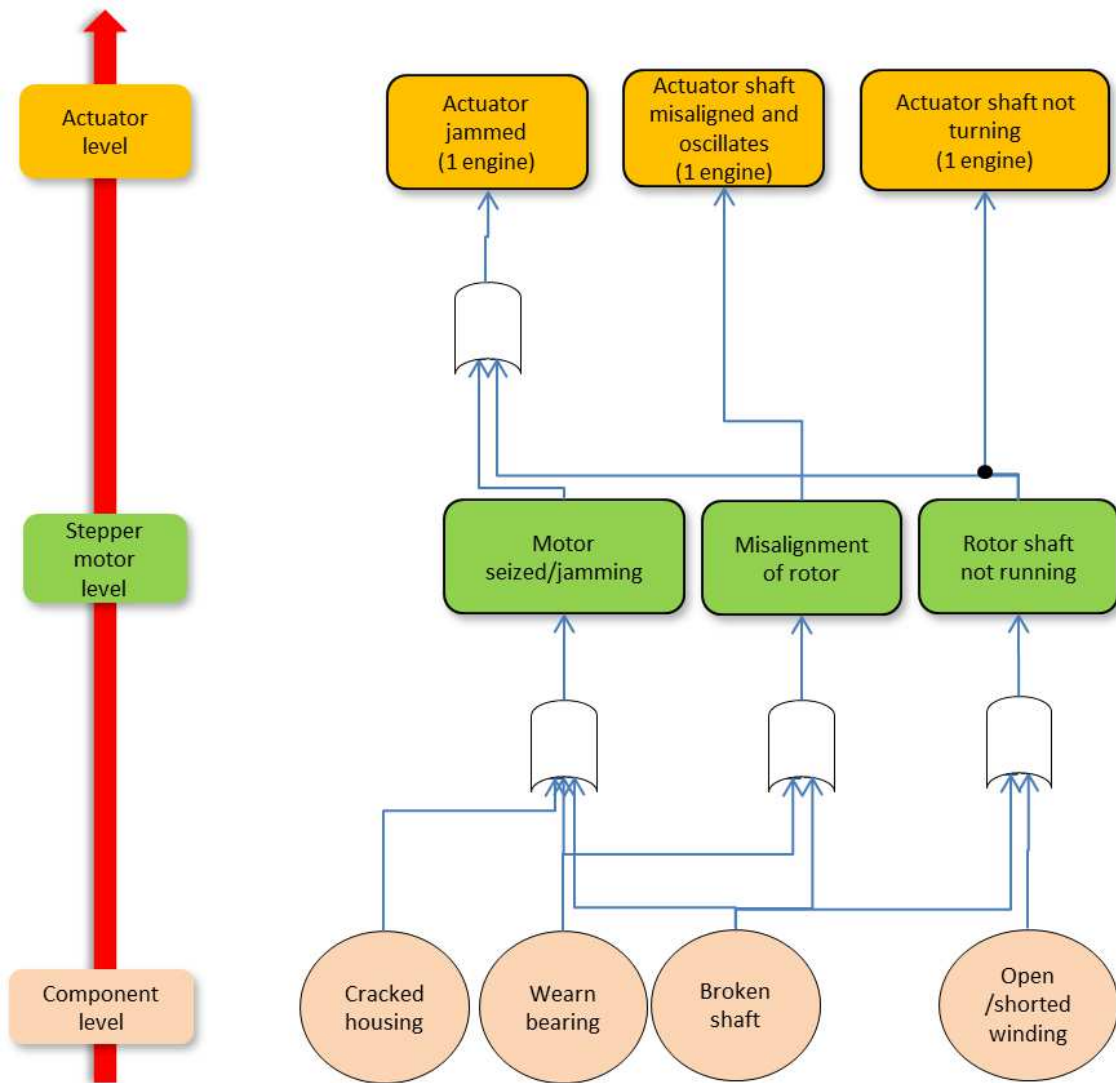


Figure 2.18: Fault tree analysis of the hybrid stepper motor

Once the FTA is established, the rates of each subsystem level and at each logical AND/OR nodes can be processed. The next section describes the failure rate processing of an electric motor. Each component (mechanical or electrical) has a failure probability which is known and can be integrated in a global rate equation dedicated to the electric motor.

### 2.3.4 Failure rate processing

According to the NSWC standard [88], failure rates  $\lambda$  of electrical motors are processed with the following equations:

$$\lambda_M = \lambda_{BE} + \lambda_{WI} + \lambda_{AS} + \lambda_{ST} + \lambda_{GR} \quad (2.5)$$

where:

$\lambda_M$  stands for the total failure rate for the motor system, failures/million hours;

$\lambda_{BE}$  is the failure rate of bearings and equals to 1 failures/million hours;

$\lambda_{WI}$  represents the failure rate of electric motor windings, 6 failures/million hours;

$\lambda_{AS}$  is the failure rate of the armature shaft, 2 failures/million hours;

$\lambda_{ST}$  is the Failure rate of the stator housing, 0.001 failures/million hours.

The FTA and failure rates of the system is known which allows to realise the full functional hazard analysis (FHA) by evaluating severity and DAL levels of the system. The next section shows this evaluation for the fuel system.

### 2.3.5 Development Assurance Level assessment

Once the FTA was updated by identifying the possible effects from actuator faults to the engine failures, the severity allocation to the encountered events must be established. In this study, we were responsible of the safety analysis of the hybrid

LRU	Failure Mode	Failure rate (f/fh) for one element	Effects at LRU level	Severity
1 engine	Lowered fuel flow and valve position error	2E-05	uncommanded IFSD	MAJ
2 engines	Lowered fuel flow and valve position error on 2 sides	2E-10	double uncommanded IFSD	CAT
1 engine	pumping	3E-06	commanded IFSD	MAJ
2 engines	pumping on the two sides	9E-12	double commanded IFSD	CAT
1 engine	pumping and LOPC request from the pilot for 1 engine	3E-06	LOPC requested by pilot according to situation or not	MAJ
2 engines	pumping on 2 engines and LOPC request from the pilot for 1 engine	9E-12	Double LOPC requested by pilot according to situation or not	CAT
1 engine	non detected overflow in one fuel circuit and lowered fuel flow	9E-06	OSP UAC on one engine	MAJ
2 engines	non detected overflow in one fuel circuit and lowered fuel flow on the 2 engines	8E-11	OSP UAC on the two engines	CAT

Figure 2.19: Hazard assessment on events affecting one or two engines

stepper motor (HSM). In the Functional Hazard Assessment (FHA) phase, safety-critical events were established, according to known aeronautical actuation control systems. The updated FMEA table is given in figure 2.19. According to dependability attributes such as integrity and availability, degrees of architecture redundancies can be chosen depending on the severity of the feared event. The stepper motor of the actuator is safety critical and the FMEA and FHA shows that a fault occurrence can lead to a catastrophic (CAT) or major (MAJ) event. As shown in figure 2.20, triplex and quadruplex architectures are the most suitable candidates for high integrity related systems. Indeed, adding dissimilarity and redundancy for a given

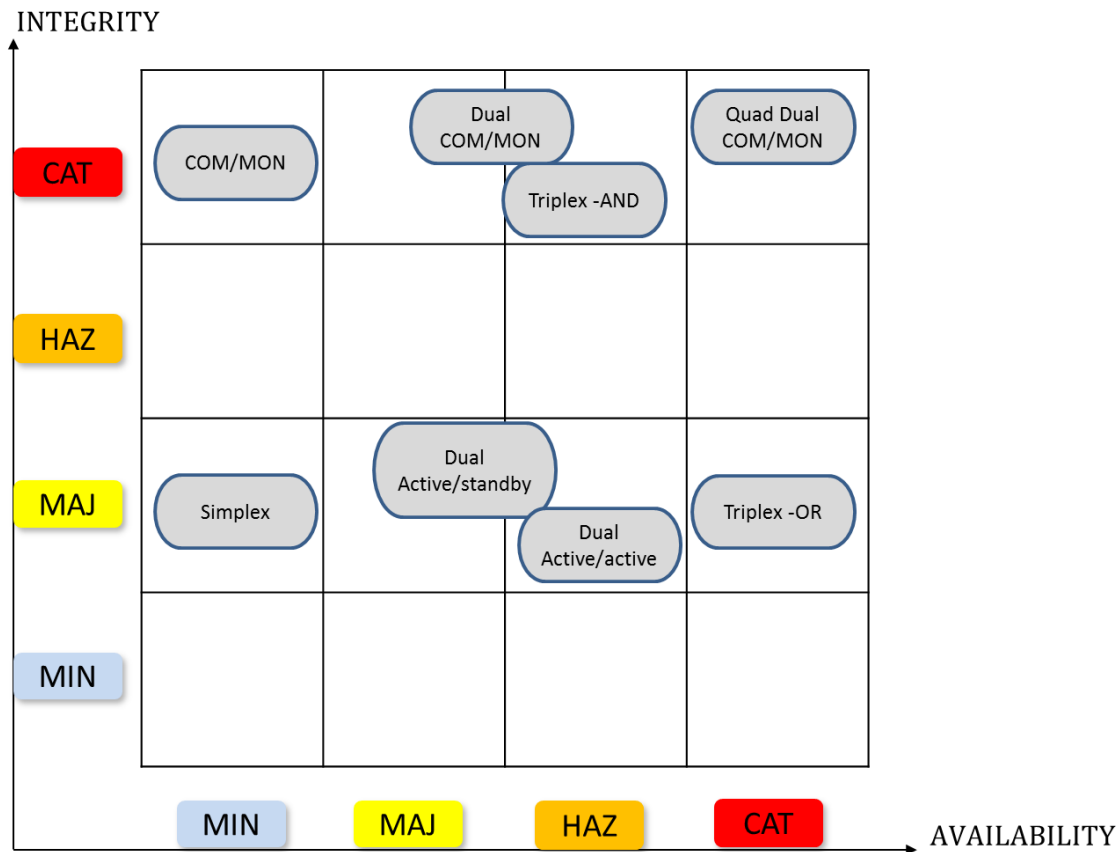


Figure 2.20: Safe architecture design

system architecture reduces the probability of a feared event. As previously seen on figure 2.7, the catastrophic and extremely improbable event is allocated to a DAL-A level, in compliance with safety standards ARP4754. The path leading to this event starts from the stepper motor of the fuel valve actuator. Therefore a DAL-A compliant redundancy architecture is required in order to minimize the CAT failure

event probability of  $10^{-9}$  failure probability per flight hour. Regarding fuel system actuators, the impact of high integrity and availability levels leads to an increase of the number of redundancies. A significant number of actuator redundancies have been realised for flight control and as an example, redundancies of flight-control actuators are compared in the next section.

### 2.3.6 Current actuator redundancies in flight control

An important number of accidents are related to loss of control in flight (LOC-I), where a technical malfunction is the initial event and responsible for loss of control. History of flight control systems [30] has shown significant improvements on flight control, especially with the emergence of fly-by-wire where flight surfaces are partially electrically supplied, reducing the number of mechanical components. Actuators monitoring is also improving due to more electrical measurements and physical and analytic redundancies.

#### 2.3.6.1 *Flight control surfaces*

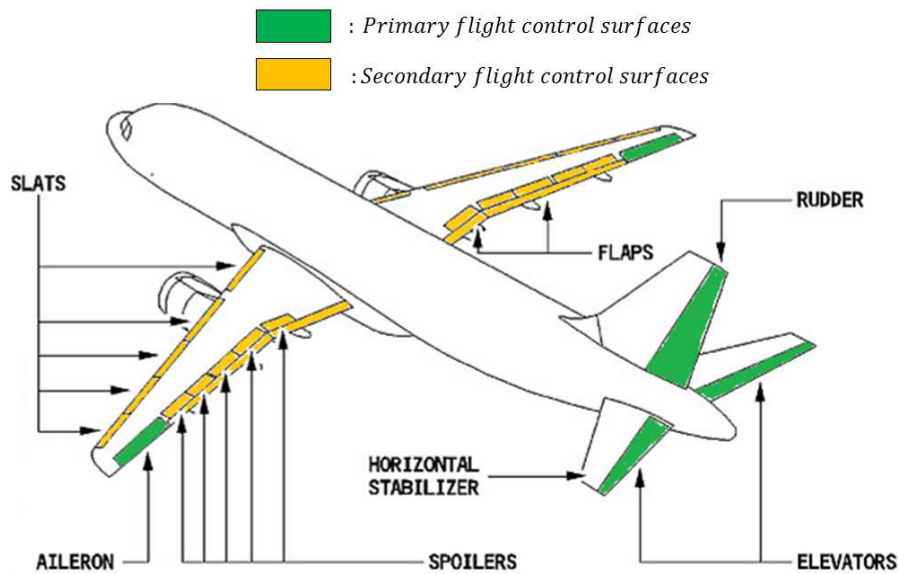


Figure 2.21: Flight control surfaces

Two types of flight control surfaces can be identified on an aircraft (Figure 2.21) :

- *primary flight control surfaces*, allowing to control the aircraft trajectory by rotating along pitch, yaw and roll axis,
- *secondary flight control surfaces*, allowing the aircraft to change its velocity during flight and landing.

These surfaces are controlled by redundant actuators and dedicated computer units shown in the next paragraph.

**2.3.6.1.1 Remark :** In a healthy flight situation, the horizontal stabilizer is considered as a secondary flight control surface. When a fault occurs on the elevators, it can be used as a primary flight control surface. This was built in the Airbus A380.

### 2.3.6.2 Flight control actuator redundancies

Actuators dedicated to the two types of flight control surfaces can also be distinguished. Figure 2.22 shows actuator redundancies for primary and secondary flight controls of the *Airbus A340* [30]. These actuators are supplied by three independent

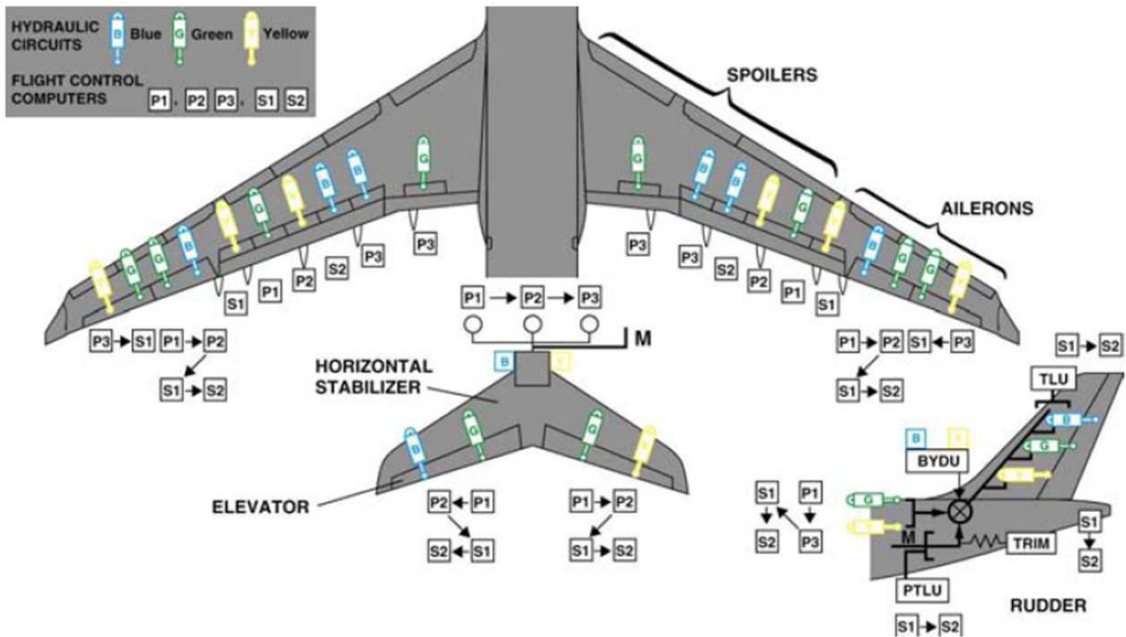


Figure 2.22: Fly-by-wire system architecture including redundancy components and reconfiguration scheme (A340), source: [30]

hydraulic circuits (blue, green and yellow) for dissimilarity and availability reasons. According to the ARP4754 standard, it is shown that several redundancy combinations are possible. The main idea of the developed patent in section 2.4 is based on this property. A safe architecture is not unique but has to be compliant with safety requirements. In this section, two existing redundancy architectures which are currently used for flight control actuators are presented, the *triplex-AND voter* and the *Dual Active/Passive* architecture. According to the ARP4754 standard, it is shown that several redundancy combinations are possible.

### 2.3.6.3 Triple Modular Redundancy: the Triplex-AND voter

Figure 2.23 shows an example of a two out of three (2 oo 3) triplex-AND voting architecture. In this case, two actuator outputs are compared to a failure threshold. When the output signal overtakes the threshold, the fault is detected. For system integrity reasons, the fault still needs to be localised. Therefore, a third actuation channel is required at least. The outputs are compared two by two allowing the identification of the faulty channel. Other voting systems including more than three channels exist such as the two out of five voters (2 oo 5).

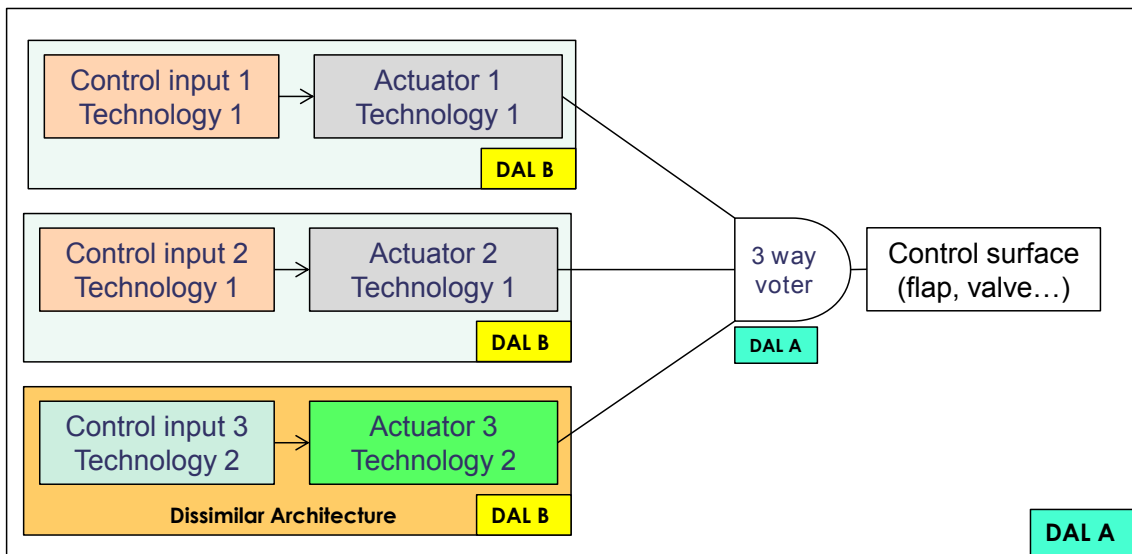


Figure 2.23: Triplex-AND voter

Failure Condition(FC) Severity	Function	Single error may induce the FC	Single error cannot induce the FC (independence)		
			2 possibilities		Remaining
Catastrophic	Critical	A	A&C	B&B	C
Hazardous	Essential	B	B&D	C&C	D
Major		C	C&E	D&D	E
Minor	Non-essential	D	D&E		

Figure 2.24: DAL allocation requirements of the ARP4754 standard

### 2.3.6.3.1 Safety and availability requirements

Safety requirements expressed by the Development Assurance Levels (DAL) (Figure 2.24) must be taken into account in the design of a triplex-AND voter. In order to ensure the required severity level DAL-A on the controlled surface system (including the control surface, the voter and the three actuators), the control input generation block and the corresponding actuator must be dissimilar (no common modes) and DAL-B compliant, as shown in figure 2.23. Therefore, the three actuation channels must be DAL-B compliant to ensure the global DAL-A level of the triplex-AND voter. Dissimilar designs also require different technologies in software and hardware development. Availability is satisfied in this case by the presence of a minimum of two actuators. If the main actuator fails, a back-up actuator is available.

The next solution proposes a flight control design based on two redundant actuators.

### 2.3.6.4 Quadriplex-dual redundant actuator architecture

The presented configuration is based on a AIRBUS Common(COM)/Monitoring (MON) (Figure 2.25). Four control channels are controlling two physical actuators (Quadriplex-dual) which are driving the same surface. Each actuator is independent and monitored by algorithms located either in the ECU or FCU, or directly in the Actuator housing. For the same dissimilarity reasons as the triplex-AND voter, no common modes are tolerated, hardware and software parts are developed with different technologies. Also, a multiple state control switch permits, for a determined threshold value, the monitoring unit to disable the main drive unit and switching

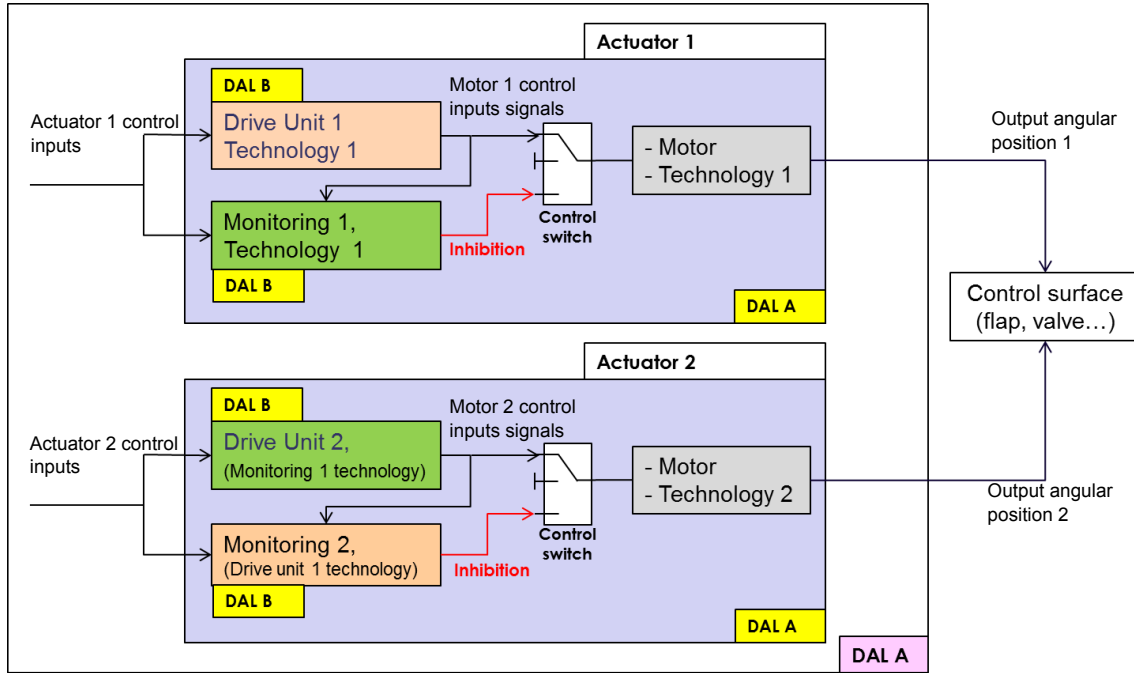


Figure 2.25: Quadriplex-dual redundant actuators

the main control path to a downgraded backup mode.

The global DAL A is satisfied with a combination of dissimilar DAL B on COM and MON blocks. Monitoring and drive unit functions are based on the same properties but with different technologies.

Most of these algorithms are based on linear dynamical systems and can not be rid of model uncertainties due to the real nonlinearity of the actuator. Dissimilarity in this type of architecture is necessary in order to be compliant with integrity and availability requirements.

The next proposed monitoring architecture is based on the previous material and physical redundancies and the ARP4754 standard.

## 2.4 Proposal of a safe and robust architecture

The previous seen quadriplex-dual redundant actuator architecture presents major drawbacks such as:

- a significant increase of weight, size, power and cost due to the redundancy of physical components,



- an increase of system complexity,
- a fault occurrence implying the direct inhibition of one actuator reduces evidently the availability level of the flight or engine control function.

The proposed architecture (Figure 2.26) shows improvement on the previous multiple redundant systems. The motor of the actuator is monitored by a dedicated

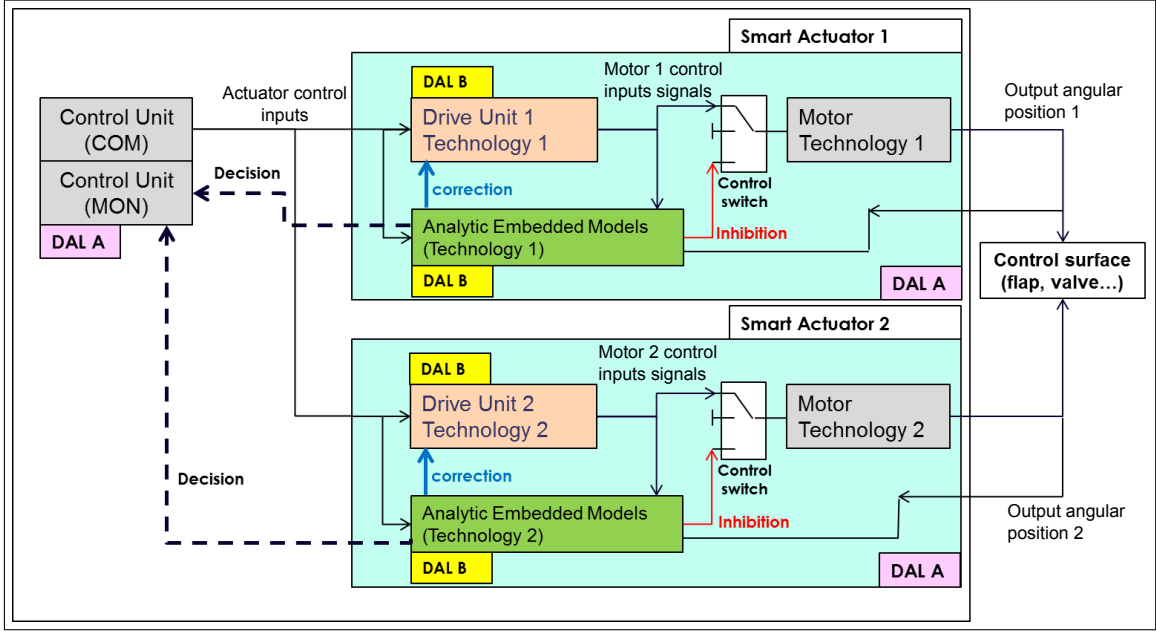


Figure 2.26: Suggested actuator redundancy architecture

analytic embedded (AEM) model which is a software program and also used for analytic redundancy. The model algorithm allows to correct the control input by sending a correction signal in order to maintain the actuator availability level in the case of a fault occurrence. If the acceptability range is overtaken, the actuator is finally inhibited. Embedding such analytical functions imply major improvements regarding:

- **Availability preservation** : Correcting the actuator input by the decision of the analytic model allows to maintain the system availability if a fault occur instead of inhibiting the main actuation channel.
- **Sise, Weight and Power (SWaP) and cost decrease** : The proposed monitoring functions can be embedded directly on the FCU, the ECU or on a

local computer unit. Instead of using physical components for system redundancy, software functions are used in this case. On current quadriplex-dual redundant architectures there would be a win of two physical monitoring channels. Instead of correcting by switching the faulty actuator "OFF", the motor control voltage is adjusted by a correction signal sent by the analytic embedded model.

- ***Servicing and maintenance decrease*** : By correcting the control input signal on a fault occurrence, the maintenance frequency on the actuator channels could decrease, allowing to shorten aircraft ground time, which also implies an important cost reduction for airline companies. This solution also allows to improve predictive maintenance and reduce corrective maintenance by its capability of recording corrected faults events. The system awareness is thereby more reliable and efficient.
- ***Integrity improvement*** : The dissimilarity of the proposed architecture reduces the presence of common modes reducing therefore the probability of faults occurrences. Regarding the computer unit, the software must be designed by two separate teams. Each actuator has its own dedicated monitoring unit which is embedded directly with the actuator and also different for each actuator (they are therefore called smart actuators). All software and hardware components must be of different technologies to maintain integrity.
- ***Monitoring robustness*** : Monitoring algorithms are based on input reconstruction by endogenous feedback developed in Chapter 1. As explained in the previous chapter, the actuator model is linearised with an exact linearisation method which was proven to be stable.

Also, the use of linear equations in the algorithm allows to ease measuring noise attenuation which implies a more accurate fault diagnosis. The resulting equivalent linear model is then used in the monitoring function of each actuator.

This architecture proposal has led to a patent deposition with THALES Systèmes Aéroportés, the IMS Laboratoy (Université de Bordeaux) and the ESTIA Recherche Laboratory (Bidart) [13].

To conclude on this innovation, the proposed architecture allows to reduce the number of physical redundancies without downgrading the safety of the architecture. This is possible because of the monitoring models used in analytic redundancies. The term "*analytic sensor*" also designate the developed models. Originally, this architecture was designed regarding aviation regulation standards ARP4754 but it was also patented for each industrial domain dealing with critical systems. As perspectives for analytic embedded models (AEM), several functions related to fault tolerant control could be developed such as fault recovery, mechanism reconfiguration, health monitoring and prognostic. In order to reduce the complexity of aircraft certification processes, AEM could be able to record the correction information before looping it with the actuator control input.

## 2.5 Conclusion

In this chapter the design of a monitoring system for critical systems was described. First, safety critical systems were presented in the case of fuel systems and flight system control of different air planes and helicopters. In the first section, definitions and safety assessment methods provided by international airworthiness organisations were given. Also, a review on current system engineering tools and methods used in industry was developed in the second subsection of this chapter. In the second section, the case study of the safety assessment of a flight critical stepper motor was realized. The analytical models developed in Chapter 1 were proposed for a safe architecture based on analytical redundancy. The Development Assurance Level of the Architecture was also discussed. To conclude, analytic redundancy is a major advantage for safety critical architectures because analytic monitoring might lead to a decrease of the number of material redundancies actually present in current flight critical architectures. On the other hand, difficulty of analytic

redundancy designs relies in the exactitude of the model of the monitored system. As described in Chapter 1, the proposed architecture was designed with a safe and robust linearisation method reducing disturbances and fault-detection thresholds. As a result of this architecture, a patent dedicated to critical systems has been submitted .

In the next Chapter, I realised a testing bench in order to be able to generate faults on a hybrid stepper motor windings of a flight critical actuator. The proposed diagnosis algorithms presented in Chapter 1 were embedded on a real-time machine, representing a Flight-critical Engine Computer Unit, as depicted in Chapter 2. The aim of this experience is to demonstrate the improvement of diagnosis results by reducing dedicated algorithms complexity.

# Chapter 3

## Experimental setup

### Contents

---

<b>3.1</b>	<b>Introduction . . . . .</b>	<b>101</b>
<b>3.2</b>	<b>Actuator model validation and integration on test bench . .</b>	<b>102</b>
3.2.1	Test bench design . . . . .	102
3.2.2	Path planning of control inputs by dynamic inversion . .	103
3.2.3	Dynamic inversion based model linearisation . . . . .	106
3.2.4	PWM signals processing for power board inputs . . . . .	108
3.2.5	Sensors measurements validation . . . . .	109
3.2.6	Stator fault generation . . . . .	113
<b>3.3</b>	<b>Diagnosis algorithms integration and validation on bench . .</b>	<b>116</b>
3.3.1	Residuals generation based on dynamic inversion and a standard Kalman filter . . . . .	116
3.3.2	Diagnosis models comparison . . . . .	121
3.3.3	Conclusion . . . . .	124

---

---

**Abstract:** This chapter is dedicated to the experimental setup of the fault diagnosis algorithms based on endogenous feedback linearisation. Within an aeronautical context regrouping an aeronautics industry, *THALES Systèmes Aéroportés* and the two laboratories: *IMS* from the *University of Bordeaux* and *ESTIA Recherche* (Bidart), a test bench was required to demonstrate the robustness and performances of fault detection algorithms applied to a hybrid stepper motor (HSM) with short-circuited windings. In the first section, the Matlab Simulink model of the fault diagnosis algorithms of the HSM will be detailed. The second section describes the design and mechanical assembly of the test bench. The last section is dedicated to the results analysis and a general conclusion will be given regarding the efficiency of exact linearisation in model-based diagnosis approaches.

---

## 3.1 Introduction

The experimental setup was realised to demonstrate on a test bench the efficiency of analytic redundancy for a faulty HSM of a flight critical actuator. The stator windings were uncoiled in order to test the behaviour of the motor when short-circuits occur. This fault type represents 40% of electric motor faults and is not negligible. An endogenous feedback linearisation of the motor model is realised in order to compare the diagnosis performances of linear and nonlinear observers such as the standard and extended Kalman filters (SKF and EKF). The linearised model will be used with the SKF and the resulting measurements estimates will be compared to the estimates of the nonlinear EKF. Diagnosis performances will be evaluated regarding the amplitudes of the detection thresholds.

In section 3.2, the chosen control method of the stepper motor based on path planning is described. The control inputs are generated by a model on a host PC which is connected to a real-time machine. Sensors calibration and faults generation is also detailed. The integration and validation steps are realised with a healthy stepper motor. In section 3.3 fault-detection based on endogenous feedback linearisation and a standard Kalman filter is presented. The measurement estimations are compared in healthy and faulty cases. Next, in order to realise a comparative analysis, the above filter is replaced by an EKF without dynamic inversion. Both filters are thus compared regarding the efficiency of residual generation.

## 3.2 Actuator model validation and integration on test bench

In order to validate the presented diagnosis algorithms with measured values, I designed and assembled the following test bench. The related specifications are given in Appendix C.

### 3.2.1 Test bench design

Algorithms are designed in the MATLAB/Simulink environment on a host PC (Figure 3.1) and then embedded on a physical real-time machine. This *Hardware In the*

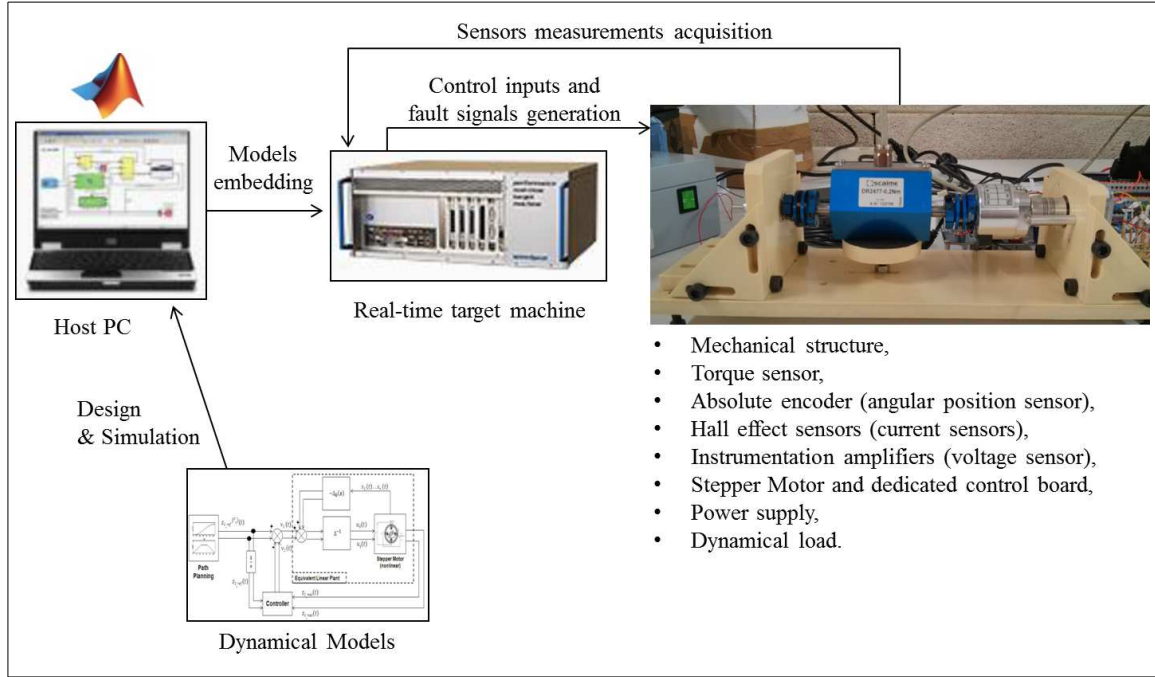


Figure 3.1: Hardware In the Loop (HIL) integration in the test bench design

*Loop (HIL)* approach allows to monitor sensor outputs and the processed variables of the dynamical models in real-time. Model parameter tuning is also possible in real time. The required communication ports between the host PC and the real-time machine, sensors and the motor control board are given in Appendix C.

The functional scheme designed for the test bench realisation is described in figure 3.2. Blocks 1, 2, 3, 13, 14, 15, 16 correspond to the developed model functions,



presented in Chapter 1. The remaining blocks represent the physical components of the test bench. First, path planning (angular position, speed, acceleration and jerk)

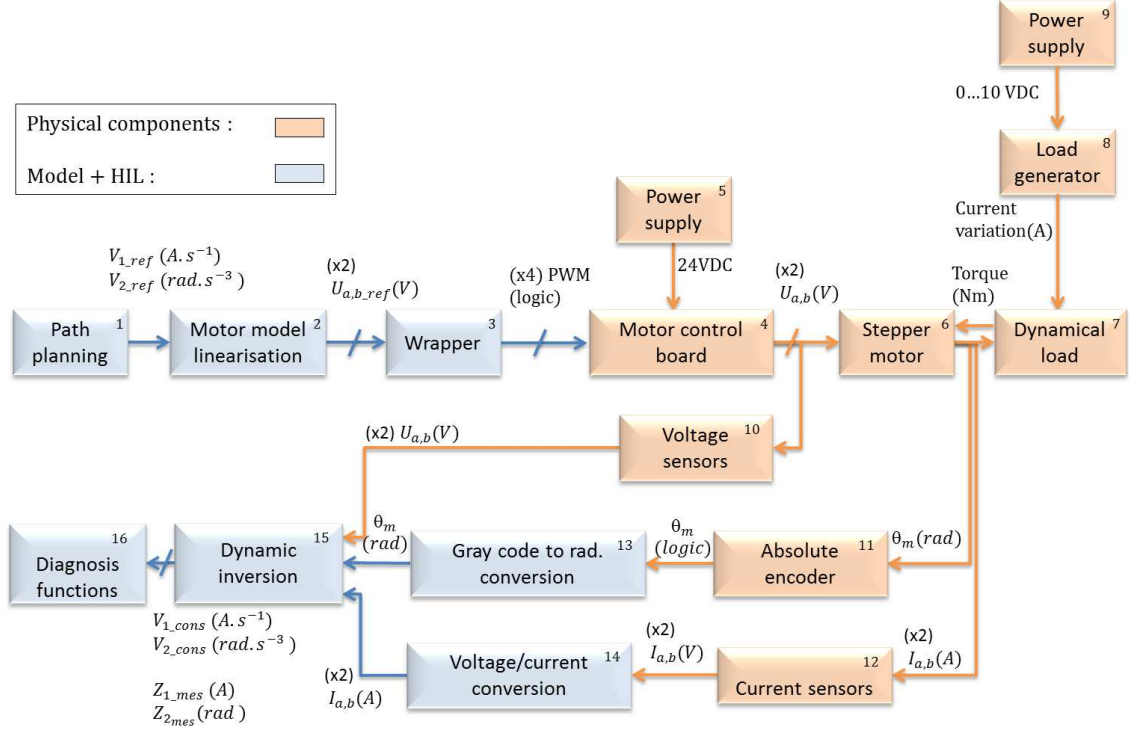


Figure 3.2: Physical components and analytical functions interactions

of the rotor angular position are processed. As a result, the motor control inputs (voltage signals) are processed according to the dynamical equations of the motor. The real time machine converts the logical PWM sequence onto physical signals which are connected to the power board of the motor. Physical signals are then measured by the sensors and finally acquired and processed by the embedded model on the real time machine. The next section describes the path planning equations enabling frequency modulation of the input signals.

### 3.2.2 Path planning of control inputs by dynamic inversion

The trajectories of angular position, speed, acceleration are obtained by integration of the angular jerk equation where the jerk must be a continuous and differentiable function. The jerk is considered as a piecewise cosine function. The maximum amplitude of the angular jerk is processed regarding maximum acceleration ( $a_{max}$ ) and

speed ( $\omega_{max}$ ) inputs. The corresponding acceleration and jerk period  $T$  is determined by tuning the smooth jerk equation ((3.1)) regarding the values of  $a_{max}$  and  $\omega_{max}$ . These particular jerk equations were chosen in order to be compliant with continuous differentiations. The angular jerk equation is given by:

$$j(t) = \begin{cases} \frac{j_{max} \left(1 - \cos\left(\frac{2\pi t}{T}\right)\right)}{2} , & \text{if } t \leq T \\ -\frac{j_{max} \left(1 - \cos\left(\frac{2\pi t}{T}\right)\right)}{2} , & \text{if } T < t \leq 2T. \end{cases} \quad (3.1)$$

The integration of  $j(t)$  gives the angular acceleration, shown in the following equation :

$$a(t) = \begin{cases} \frac{j_{max} \left(t - \frac{T \sin\left(\frac{2\pi t}{T}\right)}{2\pi}\right)}{2} , & \text{if } t \leq T \\ -\frac{j_{max} \left(t - \frac{T \sin\left(\frac{2\pi t}{T}\right)}{2\pi}\right)}{2} + j_{max}T , & \text{if } T < t \leq 2T. \end{cases} \quad (3.2)$$

The motor is supposed to start at null speed, implying :  $\omega(0) = a(0) = j(0) = 0$ .

The angular velocity is then given by :

$$\omega(t) = \begin{cases} \frac{j_{max} \left(\frac{t^2}{2} - \frac{T^2 \cos\left(\frac{2\pi t}{T}\right)}{4\pi^2}\right)}{2} , & \text{if } t \leq T \\ -\frac{j_{max} \left(\frac{t^2}{2} - \frac{T^2 \cos\left(\frac{2\pi t}{T}\right)}{4\pi^2}\right)}{2} + Tj_{max}t - \frac{T^2 j_{max}}{2} , & \text{if } T < t \leq 2T. \end{cases} \quad (3.3)$$

In order to express the trajectories  $j(t)$ ,  $a(t)$  and  $\omega(t)$  as a function of  $a_{max}$  and  $\omega_{max}$ , the maximum acceleration  $a_{max}$  is first expressed as a function of  $j_{max}$  and  $T$

according to equation (3.2) at  $t = T$  such as :

$$a_{max} = \frac{T j_{max}}{2} \quad (3.4)$$

Also, according to equation (3.3),  $\omega_{max}$  is processed at  $t = 2T$  and is given as a function of  $j_{max}$  and  $T$  by :

$$\omega_{max} = \frac{4\pi^2 T^2 j_{max} + T^2 j_{max}}{8\pi^2}. \quad (3.5)$$

The expression of  $j_{max}$  and  $T$  can be processed as a function of  $a_{max}$  and  $\omega_{max}$  by solving equations (3.4 and 3.5). Hence :

$$\begin{cases} T &= \frac{4\omega_{max}\pi^2}{a_{max}(1+4\pi^2)} \\ j_{max} &= \frac{a_{max}^2(1+4\pi^2)}{2\omega_{max}\pi^2}. \end{cases} \quad (3.6)$$

These processed trajectories are shown in figure 3.3 and were normalised for a better visibility. Using the dynamical inversion of the stepper motor model given in

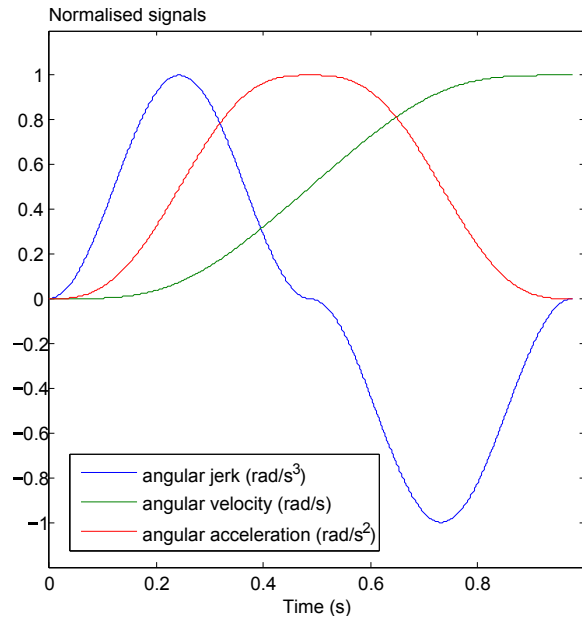


Figure 3.3: Normalised reference trajectories

Chapter 1, the voltage trajectories can be processed at desired maximum angular speed  $\omega_{max}$  and acceleration  $a_{max}$  as described in the next section.

### 3.2.3 Dynamic inversion based model linearisation

In this approach, no motor-load was considered. According to figure 3.4, the algorithm allowing to process the  $u_a$  and  $u_b$  voltage trajectories is described by the following steps:

- processing the inputs  $v_1$  and  $v_2$  of the linear equivalent model of the HSM,
- the direct current  $i_d$  is set to 0 and the quadratic current  $i_q$  is expressed by an equation containing  $\omega$ . It was shown in chapter 1 that the motor model was flat with flat outputs  $z_1 = i_d$  and  $z_2 = \theta$ ,
- the linearisation functions  $\Delta(x)$  and  $\Delta_0$  defined in Chapter 1 allow the processing of the  $v_d$  and  $v_q$  voltages which are then changed into  $u_a$  and  $u_b$  by Park transform,
- a PWM signal generation algorithm allows to generate the pulse trains of the generated  $u_a$  and  $u_b$  voltages,
- finally, the PWM pulse trains are generated by the real time machine and used directly with the power board of the stepper motor.

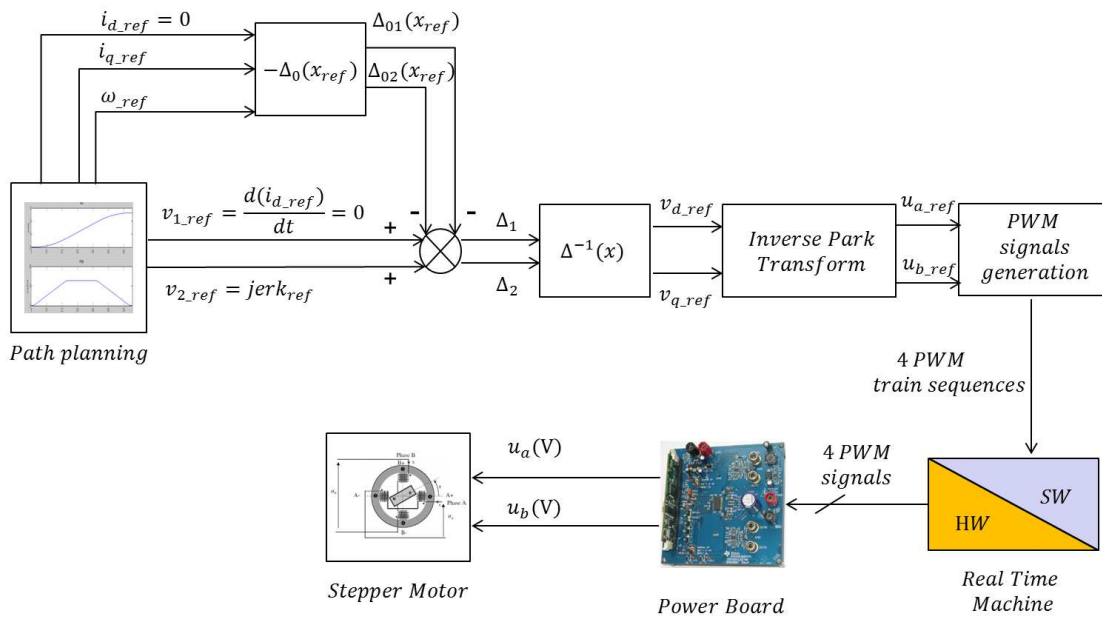


Figure 3.4: Control signals generation and linearisation

The following plots (Figure 3.5) show the resulting generated voltages and currents.

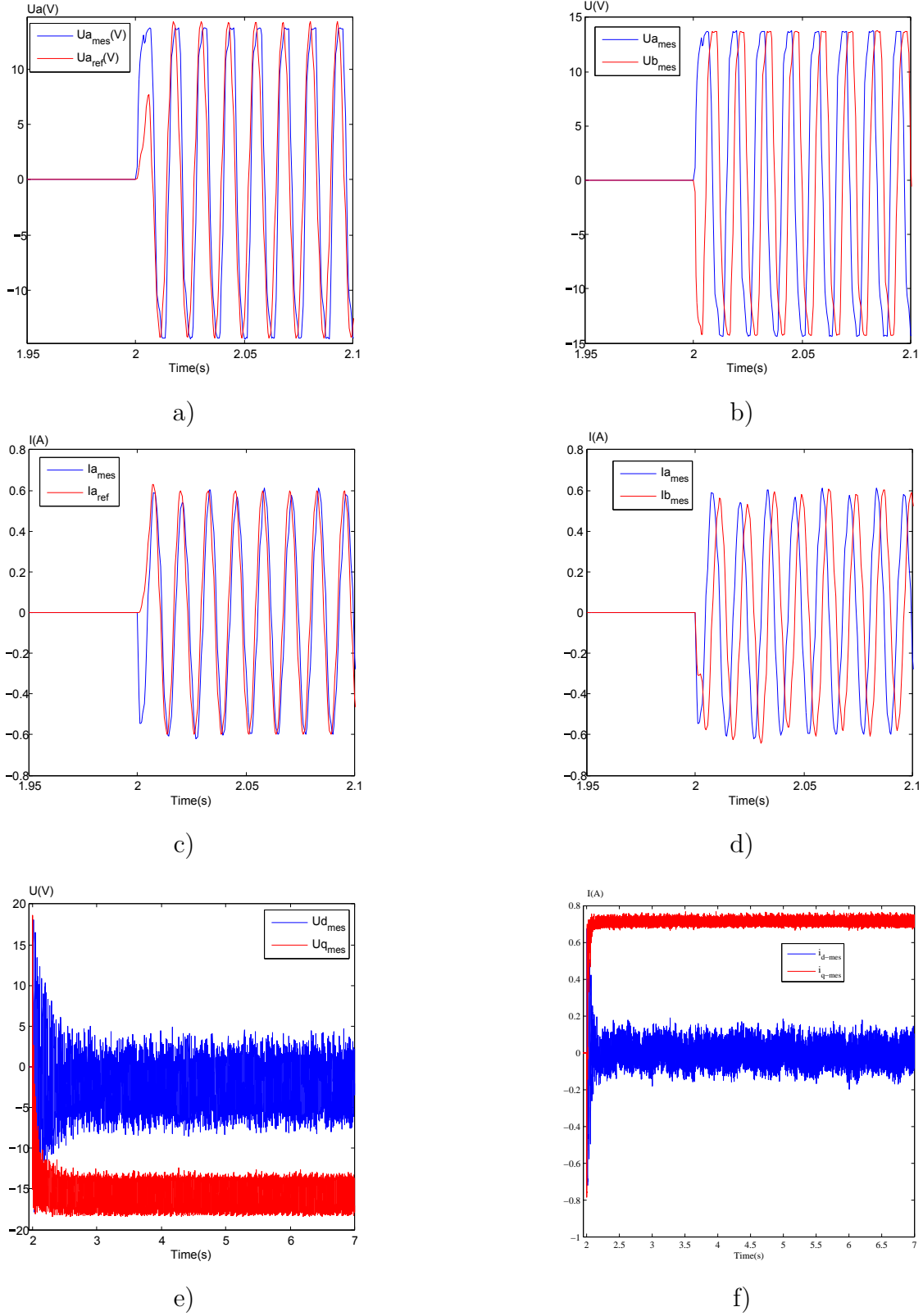


Figure 3.5: Processed and measured voltages (a,b) and currents (c,d); Measured direct and quadratic voltages (e) and currents (f)

### 3.2.4 PWM signals processing for power board inputs

The normalised voltage signal corresponds to the duty cycle variation of the Pulse Width Modulation (PWM) generator. To generate the PWM signal with the real

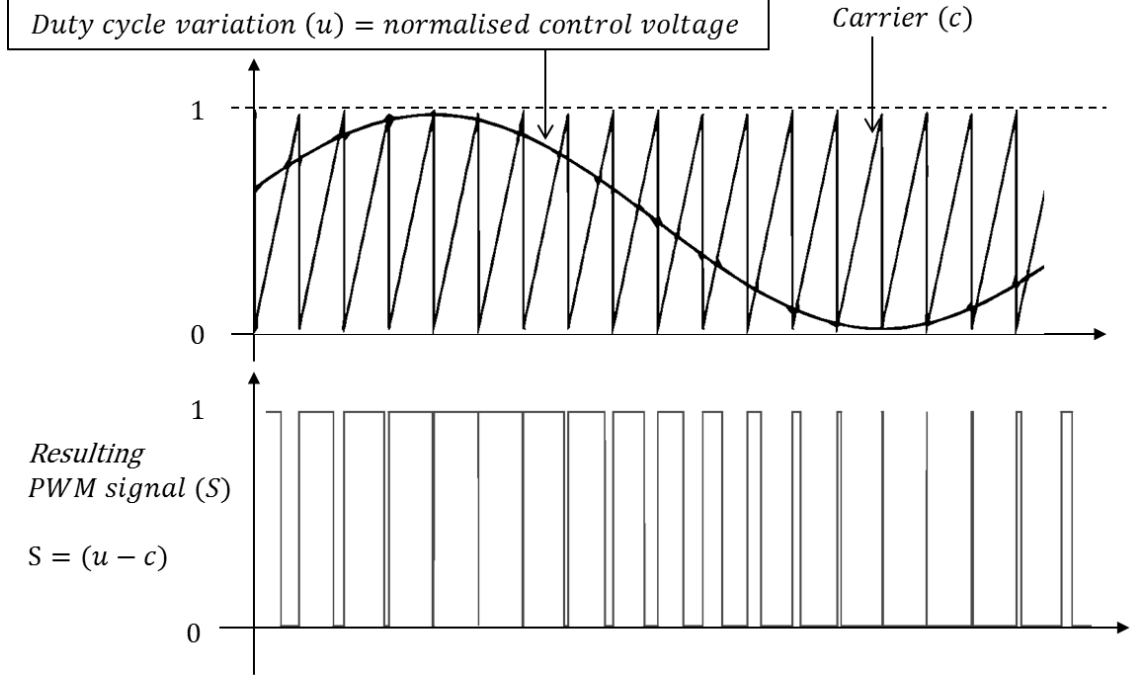


Figure 3.6: PWM signal generation with duty cycle variation

time machine, a clock compare value has to be taken into account by the following equation:

$$\text{Compare value} = \text{round} \left( \frac{\text{FPGA frequency}}{2 \times \text{PWM frequency}} - 1 \right). \quad (3.7)$$

The PWM trains are generated by the real time machine, corresponding to a digital signal with logical high and low states respectively equal to 5V and 0V. The sequence allows to switch on and off the transistors  $T$  of the double H-bridge (Figure 3.7). In order to create a positive current ( $i_a$ ) in the  $A$  phase of the motor,  $T_1$  and  $T_4$  are set to *ON* simultaneously and  $T_2$  and  $T_3$  are switched off. Controlling the HSM by PWM allows to set maximum angular speed and acceleration parameters. The frequency of the generated voltage signals is modulated with the acceleration and speed trajectories as shown in figure 3.8.

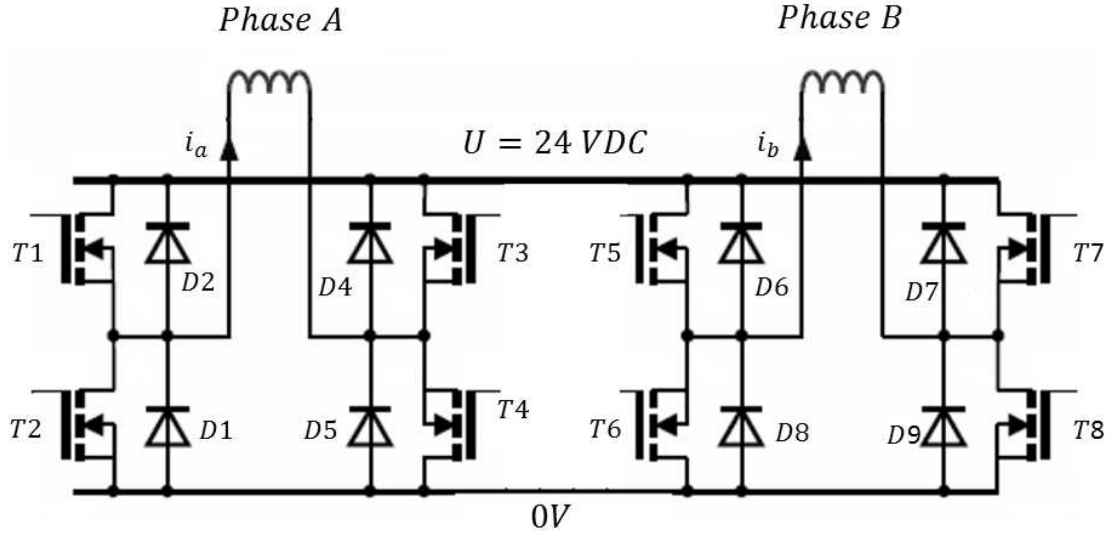
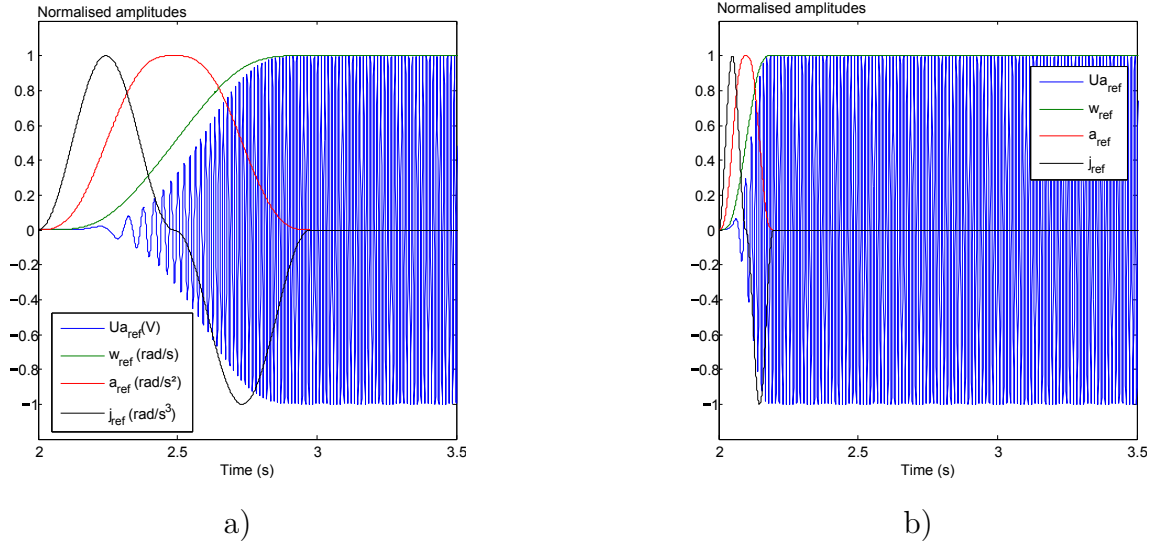


Figure 3.7: Double H-bridge of a bipolar stepper motor

Figure 3.8: Normalised  $U_a$  signal generation at  $w_{max} = 10 \text{ rad/s}$  and  $a_{max} = 20 \text{ rad/s}^2$  (a) and  $a_{max} = 100 \text{ rad/s}^2$  (b)

The next section describes the realisation of sensing functions, the required filters for noise cancellation and the set-up allowing to generate a certain percent of short circuits in a stator coil.

### 3.2.5 Sensors measurements validation

**Hall effect sensors:** To realise current measurement, Hall-effect sensors were used generating a voltage signal which is proportional to the current. The voltage

to current ratio is realised within the simulation model after signal acquisition.

**Torque sensor:** The torque constant  $K_t$  of the stepper motor model had to be confirmed by measurement. A two-shaft torque sensor was used where one shaft is connected to the load and the other to the motor shaft with flexible joints as shown in figure 3.9. To measure the torque constant, the load side is intentionally jammed

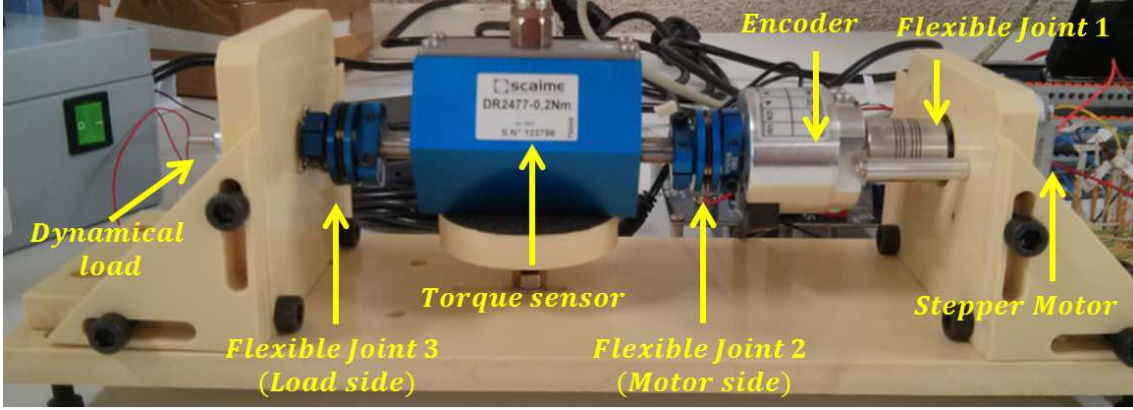


Figure 3.9: Mechanical test bench

and the motor is powered. The equation:

$$T_{max} = K_t \times I_{max} \quad (3.8)$$

is used for different values of the maximum torque  $T_{max}$  and maximum current  $I_{max}$ .

As a result, the mean value of  $K_t$  was found to be  $K_t = 0.0137 \text{ Nm/A}$ .

### 3.2.5.1 Sensor noise filtering

To measure the voltages in each phase of the stepper motor, instrumentation amplifiers are used as shown in figure 3.10. The main drawback of motor control by PWM is the resulting noise on the measured signal.

The voltage signal frequency is :

$$f_v = \frac{\omega}{2\pi}, \quad (3.9)$$

and equals  $f_v = 1.59 \text{ Hz}$ , with an angular speed  $\omega$  of  $10 \text{ rad/s}$ . In order to remove the PWM noise on the voltage measurement, two analog low-pass filters were realised,



one for each phase. The cut-off frequency  $f_c$  has to be far below the PWM signal frequency ( $f_{PWM} = 30 \text{ kHz}$ ) and is given by :

$$f_c = \frac{1}{4\pi R_1(C_3 + \frac{C_1}{2})}, \quad (3.10)$$

with:

$$\begin{aligned} C_1 &= C_2 \\ C_3 &= 10C_1 \\ R_1 &= R_2. \end{aligned} \quad (3.11)$$

Best results on the measured output voltage  $V_o$  were obtained with a cut-off frequency of  $f_c = 203.94 \text{ Hz}$  with  $C_1 = 22 \text{ nF}$ ,  $C_3 = 220 \text{ nF}$  and  $R_1 = 1.69 \text{ K}\Omega$ .

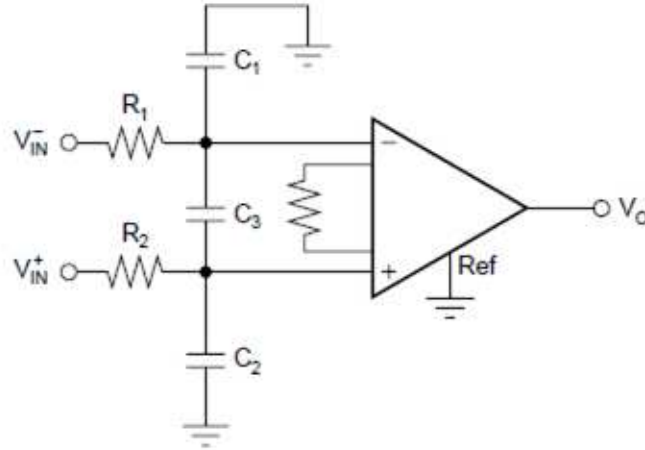


Figure 3.10: Input Low-Pass filter

### 3.2.5.2 Initial position settings

An absolute encoder (Figure 3.9) was used to measure the position of the rotorshaft. Next are given the features of the encoder :

- Supply voltage DC 7 - 30 V
- Max. current w/o load 50 mA, 100 mA
- Resolution 13 Bit

- Output: Gray code
- Drives: Clock and Data / RS422

The gray code to angle conversion is realised in the simulation model by dividing the output signal by  $2^{13}$  and then multiplying by  $2\pi$ .

The encoder does not count turns and restarts from zero when a turn is completed. In order to fit the measure signal with the processed reference signal, an unwrapping function is realised after acquiring the measurement as shown in figure 3.11.

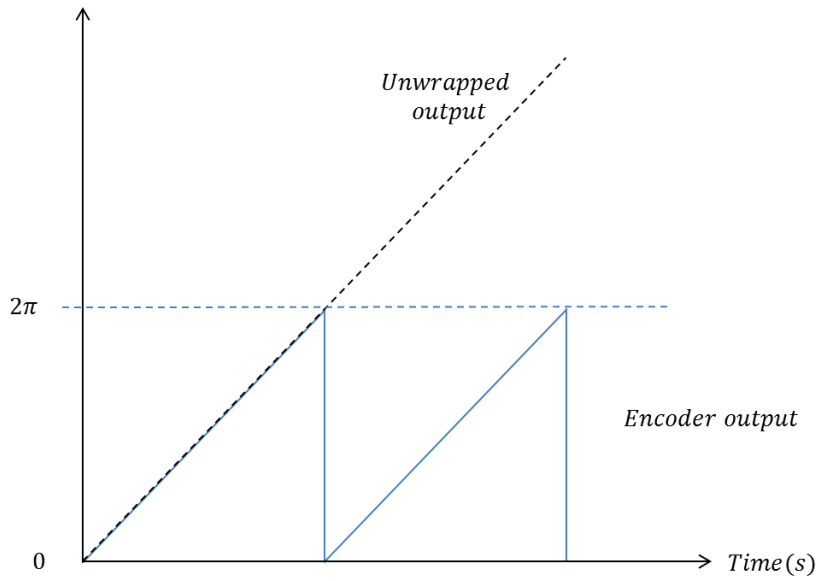


Figure 3.11: Unwrapped angle measurement

The processed reference position starts from zero radians at each new simulation. Because the encoder starts the measurement from the last position, the angular offset is measured in an initialisation phase before each simulation start and is then added to the processed angular position function. The initialisation phase is realised before each simulation run with a state chart and lasts one second. As a result the compared reference and measured angular position are shown in figure 3.12

In this section, the use and calibration of the test bench sensors allowing the real-time machine to acquire and process the measurement signals, was described. These measurements can then be processed by the embedded diagnosis algorithms developed in Chapter 1.

The generation of faults on the stepper motor is described in the next section.

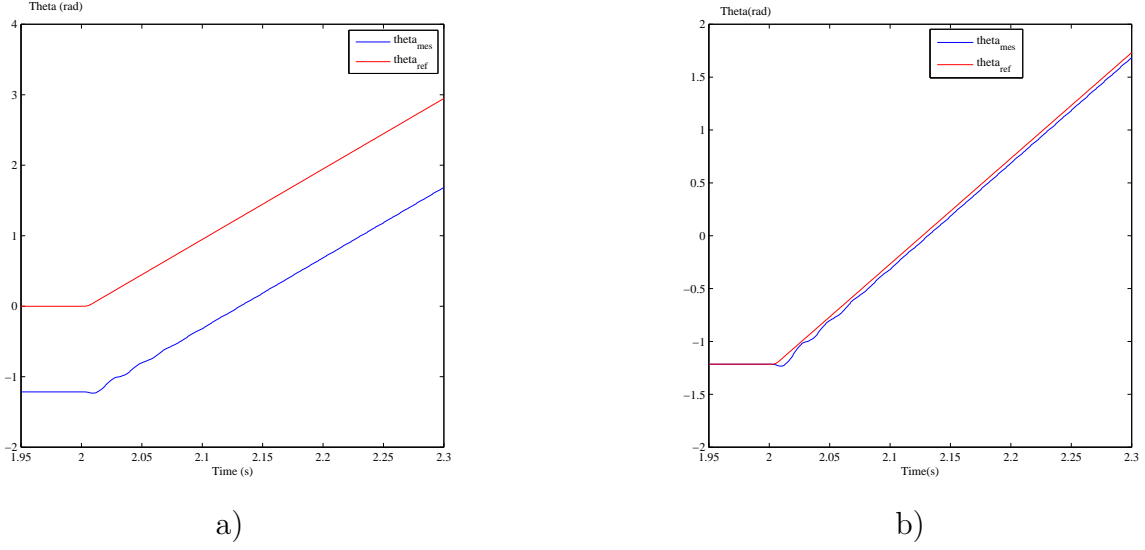


Figure 3.12: Measured and processed angular position, without (a) and with (b) position initialisation

### 3.2.6 Stator fault generation

The generated faults are shortened windings events. The stator coils of the stepper motor was undone and rebuilt to allow controllable switches (see Appendix C) to shorten a certain percentage of wires in one phase. As a result, a current increase is expected.

#### 3.2.6.1 Realisation of a shorted stator winding

To realise the new stator windings for short-circuits tests, a hybrid stepper motor was uncoiled and rebuilt with unconnected wires (figure 3.13). The stator coils are connected with switches which will be turned on and off depending on the expected shorted coil percentage as shown in figure 3.14.

#### 3.2.6.2 Fault scenarios

Each shortened coil correspond to 8 % of the global phase coils. The test is realised according to the following table:

The state "0" of a switch correspond to the "OFF" state where the circuit is open. Logically, "1" correspond to the "ON" state and the circuit is closed. In the proposed

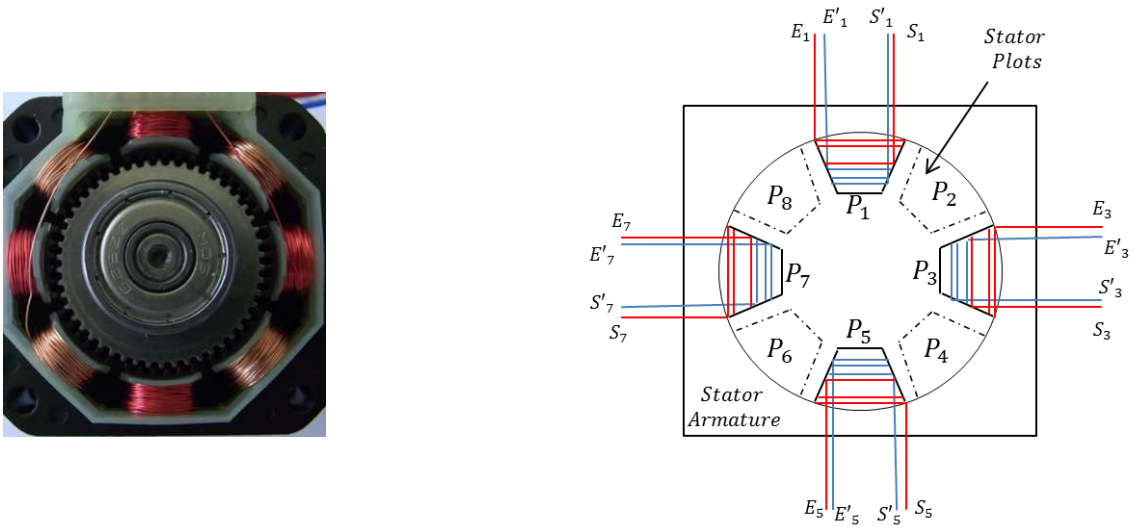


Figure 3.13: On the left-hand side a two phased stator with 8 plots (4 per phase). On the right hand side the uncoiled stepper motor for short circuits testing.

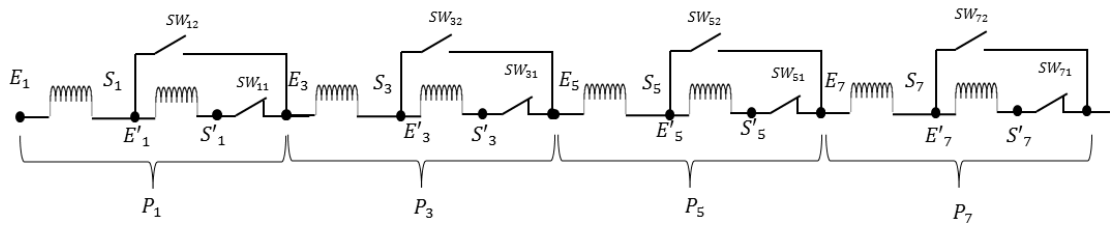


Figure 3.14: Short-circuit wiring scheme of one stator phase

Scenario configuration	Switches								Percentage of shortened coil
	SW11	SW12	SW31	SW32	SW51	SW52	SW71	SW72	
1	1	0	1	0	1	0	0	1	8%
2	1	0	1	0	0	1	0	1	16%
3	1	0	0	1	0	1	0	1	24%
4	0	1	0	1	0	1	0	1	32%

Figure 3.15: Stator fault scenarios

configuration, switches "SWii" and "SWij" are always activated in an opposite way such as there is no open circuit in the stator phase.

Also, the switches were selected in order to be:

- able to let the current flows in both ways because the stepper motor is bipolar,
- able to support about 2.5A peak (defined by the motor power board specifications),
- controllable with logical voltage states (0/5V).

Controllable *Reed relays* were therefore chosen to realise this task and are controlled by the real-time machine, where the logical ON/OFF sequence is defined in the embedded simulation model.

A coil is model as an RL circuit. When a short circuit occurs, the number of coil turns is reduced. The resulting winding has its resistance  $R_w$  multiplied by the percentage of shortened turns and the resulting inductance  $L$  is multiplied by the square of the same percentage. The wire resistance  $R_w$  is given by:

$$R_w = R_0 \frac{l}{S} \quad (3.12)$$

where  $R_0$  is the resistivity of the material (Copper for example) in (*Ohm.m*),  $l$  is the length of the wire in meters and  $S$  is the section of the wire in square meters. If the number of wires is reduced by its half, the resistance is divided by two because the length  $l$  is divided by two.

The inductance  $L$  is given as a function of the number of turns  $n_L$  and the reluctance  $\mathfrak{R}$  of the material in which the magnetic flow  $\psi$  evolves (equation 3.13)

$$L = \frac{n_L^2}{\mathfrak{R}}. \quad (3.13)$$

So if  $n_L$  is reduced by 50%, the resulting inductance  $L_r$  is equal to

$$L_r = 0.25L. \quad (3.14)$$

In this section the experimental set-up was described and model generated signals based on dynamic linearisation were validated with the corresponding sensor measurements. To realise the faults testing scenarios, a stepper motor was uncoiled allowing controllable switches to shorten a specific percentage of a stator phase windings.

The next section describes the integration and validation of diagnosis algorithms presented in Chapter 1. Two nonlinear observing methods are compared. At first, a

residual generation method is proposed, based on a standard Kalman Filter and the endogenous feedback linearised stepper motor model. Residuals and detection responses are then compared to a nonlinear Extended Kalman Filter (EKF) regarding diagnosis performance indicators.

### 3.3 Diagnosis algorithms integration and validation on bench

The proposed diagnosis method is realised regarding the direct current estimate  $\hat{i}_d$ . According to observer-based residual generation (Chapter 1, Section 1.3), the residual results from the comparison between the measurement and the measurement estimate.

#### 3.3.1 Residuals generation based on dynamic inversion and a standard Kalman filter

The linear equivalent model of the stepper motor was determined in Chapter 1 by considering the nonlinear flat system :

$$\left\{ \begin{array}{lcl} x_1 & = & z_1 \\ x_2 & = & \frac{J_m}{K_t} \ddot{z}_2 + \frac{B}{K_t} \dot{z}_2 \\ x_3 & = & \dot{z}_2 \\ x_4 & = & z_2 \\ u_1 & = & Rz_1 + L\dot{z}_1 - \frac{nLJ_m}{K_t} \dot{z}_2 \ddot{z}_2 - \frac{nLB}{K_t} (\dot{z}_2)^2 \\ u_2 & = & \ddot{\ddot{z}}_2 \frac{LJ_m}{K_t} + \ddot{z}_2 \frac{LB + RJ_m}{K_t} + \dot{z}_2 \left( \frac{RB}{K_t} + K_t \right) - nL\dot{z}_2 z_1 \end{array} \right. \quad (3.15)$$

where the flat outputs are  $(z_1, z_2) = (i_d, \theta_m)$ . It was shown that there exist an invertible function  $\beta(x)$  and a matrix  $\alpha(x)$  such as the linear equivalent model of system (3.15) is given by:

$$v = \beta^{-1}(x) [u - \alpha(x)], \quad (3.16)$$

implying

$$v = \beta^{-1}(x)u - \beta^{-1}(x)\alpha(x). \quad (3.17)$$

Equation (3.17) can then be expressed as:

$$\Delta_0(x) + \Delta(x)u = v. \quad (3.18)$$

The Brunovsky form of equation (3.18) is given by:

$$\Delta_0(x) + \Delta(x) \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} z_1^{(\rho_1)} \\ z_2^{(\rho_2)} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (3.19)$$

where  $\rho_1 + \rho_2 = n$ .

Replacing  $\rho_1$  and  $\rho_2$  by their values implies the following linear system:

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{s} & 0 \\ 0 & \frac{1}{s^3} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}. \quad (3.20)$$

The linearisation was applied to the test bench motor as depicted in figure 3.16.

Inputs  $(v_1, v_2)$  were processed with the measurements outputs of the sensors in order to realise a linear Kalman filter based on the linearised stepper motor model.

### 3.3.1.1 Validation of linear system inputs reconstruction

The reconstructed inputs  $(v_1, v_2)$  corresponding respectively to the time derivative of  $i_d$  and the angular jerk (which is the third order time derivative of the angular position  $\theta_m$ ), are shown in figure 3.17. Reconstructed  $(v_1, v_2)$  inputs are compared to their corresponding signals obtained by time derivatives. The test starts at  $t = 2$  seconds. The observed peak (Figure 3.17-a) is related to the acceleration of the stepper motor (equation (3.15)). Indeed the current increases to generate a starting torque, enabling the motor to go from the steady state to a constant angular speed. When the continuous state is reached, the angular acceleration is equal to

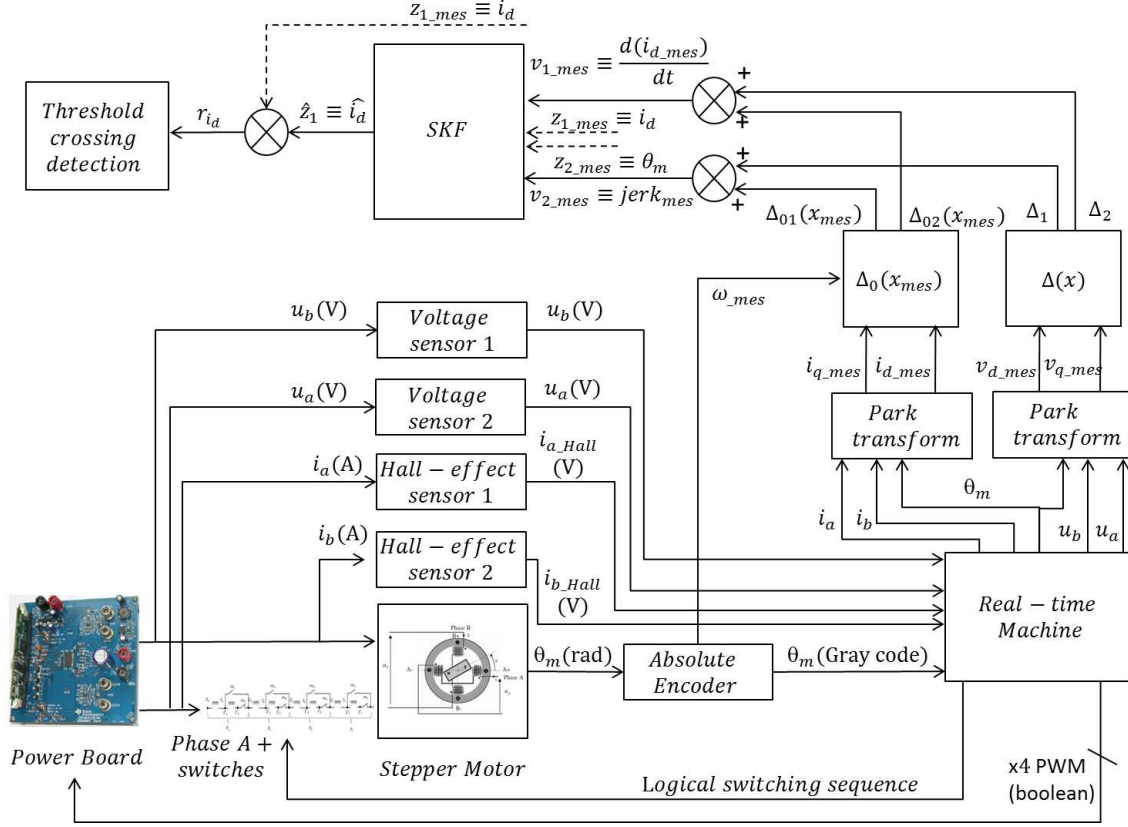
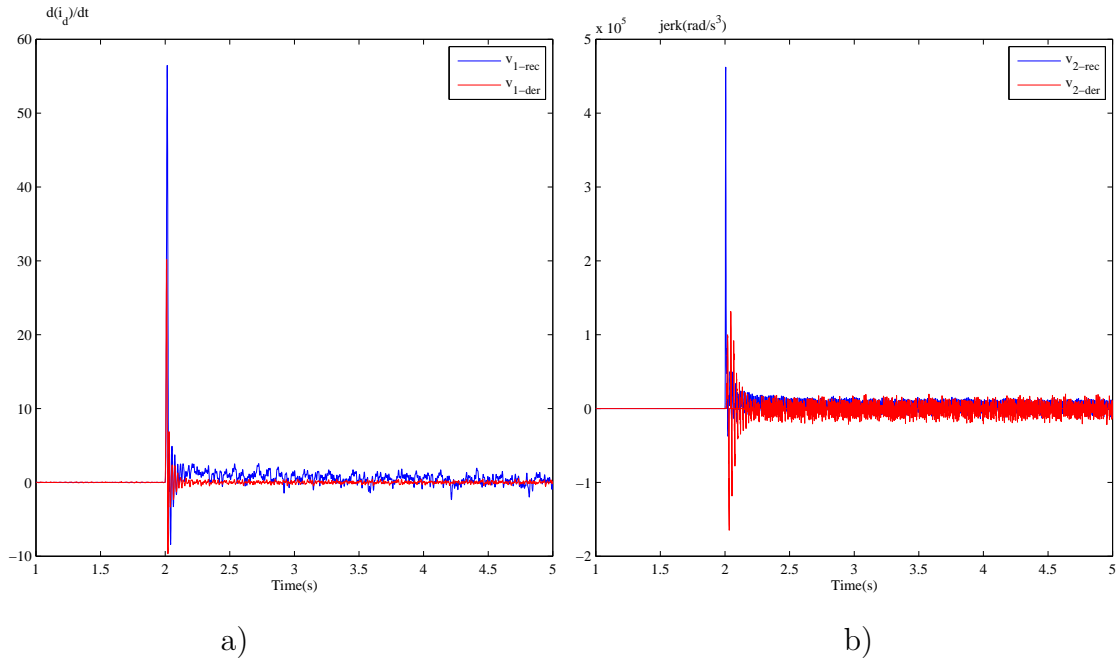


Figure 3.16: Input reconstruction of the linearised model


 Figure 3.17: (a) Reconstructed input  $v_1$  and derivative of  $i_d$  comparison ; (b) Reconstructed input  $v_2$  and  $3^{rd}$ -order derivative of  $\theta_m$  comparison

zero implying that  $v_1$  also converges to zero in continuous state. The angular jerk (figure 3.17-b) varies also from steady state to continuous state when constant speed



is reached and the signal converges to zero.

### 3.3.1.2 Measurement estimates validation

To estimate the measured  $i_d$  current, the following state ( $A$ ), control ( $B$ ) and measurement ( $C$ ) matrices were used by the standard Kalman filter, with the state space system given by equation (3.20) and the state vector  $X = (z_1 \ z_2 \ \dot{z}_2 \ \ddot{z}_2)^T$ :

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \quad (3.21)$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}. \quad (3.22)$$

The state and measurement covariance matrices ( $Q, R$ ) were tuned in order to optimise the time response of the measurement estimation and the noise amplitude with the following matrices:

$$Q = \begin{pmatrix} 0.05 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad R = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (3.23)$$

The resulting measurement estimate of the direct current  $i_d$  is shown in figure 3.18-a. The residuals of  $i_d$  shown in figure 3.18-b, reveal a mean change at  $t = 6$  seconds, corresponding to the fault occurrence. It is observable that the higher the percentage of shorted coils, the greater the amplitude of the current. Indeed, when a short circuit occurs, the resistance of the phase winding decreases suddenly which generates a current increase. These results will be compared with an extended Kalman filter (EKF) in the next section where the inputs are the quadratic and direct voltages ( $v_d, v_q$ ) obtained with a Park transform of the measurements of  $(u_a, u_b)$ , and

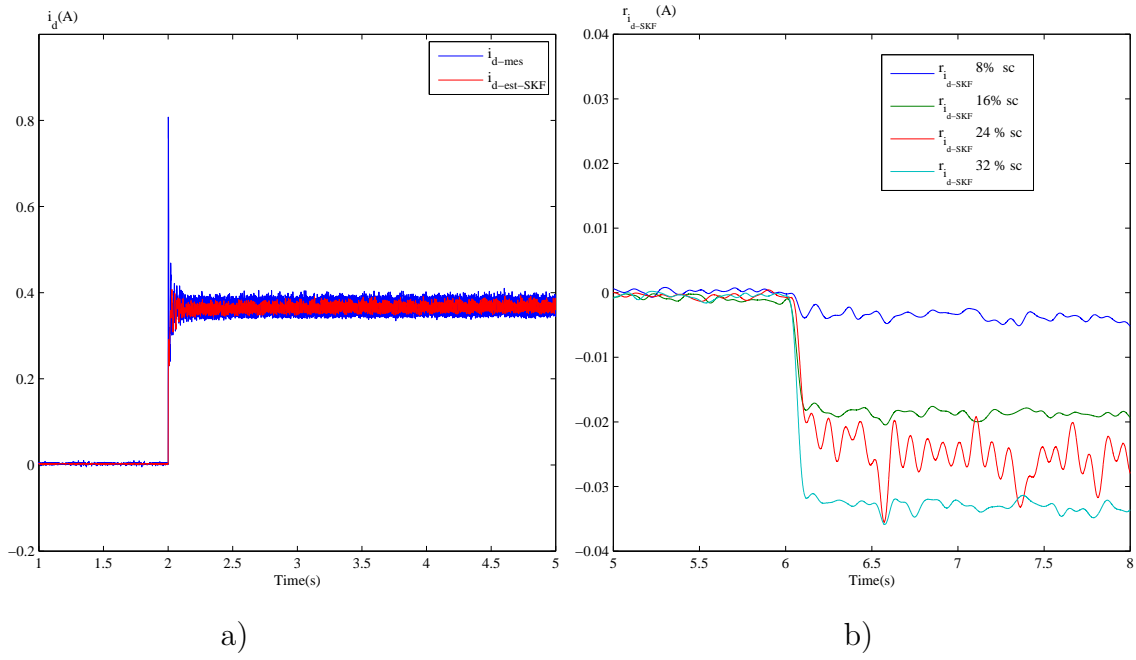


Figure 3.18: Estimated and measured  $i_d$  current in healthy mode (a); Residuals of  $i_d$  at different percentages of shorted stator coils (b)

the measurements of flat outputs ( $i_d, \theta_m$ ) as shown in figure 3.19.

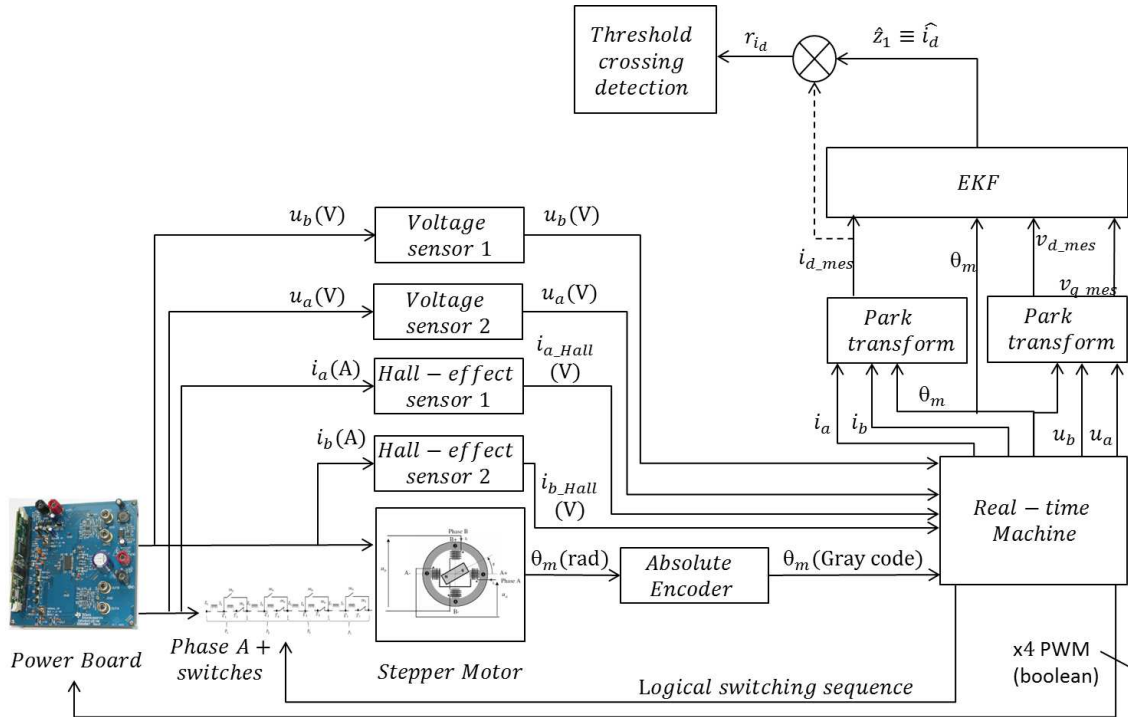


Figure 3.19: Residual generation based on the EKF

According to the recalled nonlinear model (3.24) of the HSM:

$$\begin{cases} \dot{x}_1 &= -\frac{R}{L}x_1 + nx_3x_2 + \frac{1}{L}u_1 \\ \dot{x}_2 &= -\frac{R}{L}x_2 + nx_3x_1 - \frac{K_t}{L}x_3 + \frac{1}{L}u_2 \\ \dot{x}_3 &= \frac{K_t}{J_m}x_2 - \frac{B}{J_m}x_3 \\ \dot{x}_4 &= x_3 \end{cases}, \quad (3.24)$$

the Jacobian matrices ( $A_j$ ) and ( $H_j$ ) are given by:

$$A_j = \begin{pmatrix} -\frac{R}{L} & n\omega & ni_q & 0 \\ -n\omega & -\frac{R}{L} & -ni_d + \frac{K_t}{L} & 0 \\ 0 & \frac{K_t}{J_m} & -\frac{B}{J_m} & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad H_j = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3.25)$$

and the state and measurement covariance matrices ( $Q_2, R_2$ ) are given by:

$$Q_2 = \begin{pmatrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad R_2 = \begin{pmatrix} 0.0001 & 0 \\ 0 & 0.0001 \end{pmatrix} \quad (3.26)$$

Performance of threshold-crossing based detection will be discussed regarding false-alarms and detection speed trade-offs in the next section.

### 3.3.2 Diagnosis models comparison

The residuals generated by the SKF (after dynamic inversion) and EKF are compared for different percentages of shorted stator windings.

#### 3.3.2.1 Residual thresholds crossing

In this case, a fault is generated at  $t = 6$  seconds. To detect fault occurrences, alarms are generated by threshold-crossing. Thresholds were determined by observation

in order to have the shortest time response and a minimum of false alarms. It is

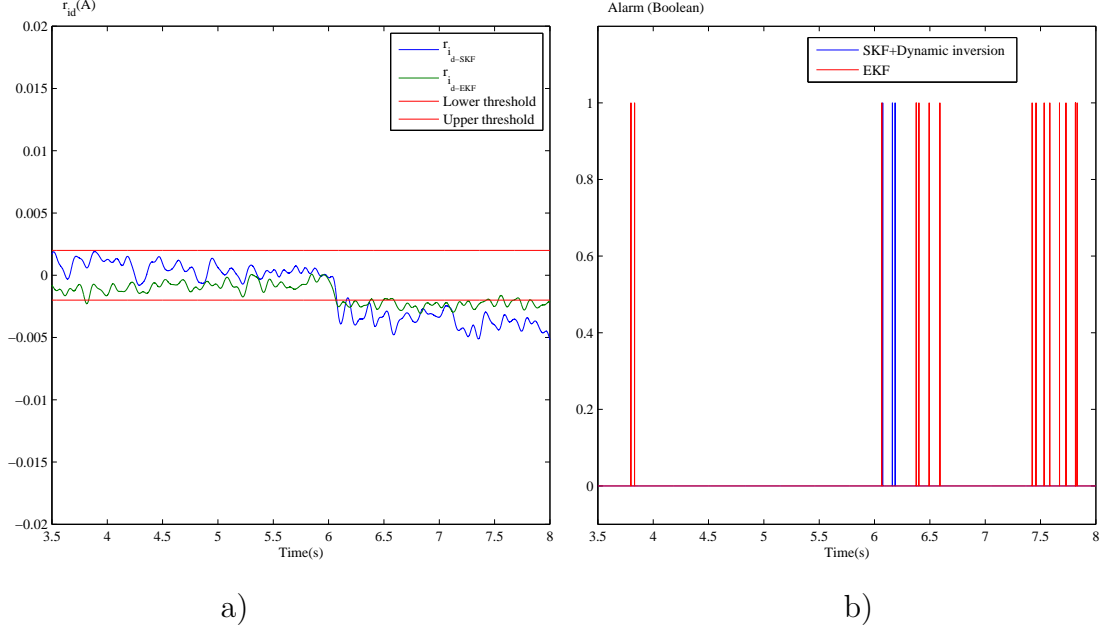


Figure 3.20: Residual (a) and alarms (b) generation for 8% of shorted windings in one stator phase

observable in figure 3.20-b that false alarms occur at the lowest percentage of shorted windings in the case of the EKF, due to a lower zero-convergence of the residual. For all cases, the shortest time response of fault detection is obtained with the SKF with dynamic inversion and linearisation of the model. Regarding non-detection, the lower the amplitude of the residual on fault occurrence, the greater the risk of non-detection. Indeed, if a greater threshold is selected, and the amplitude of the residual is below the threshold, no alarm will be generated. For all fault cases, the residual obtained by dynamic inversion and SKF has the greatest amplitude on a fault event allowing a better adaptability regarding time response and non-detection trade-offs as shown in the figure 3.22. The coloured cells correspond to the best result.

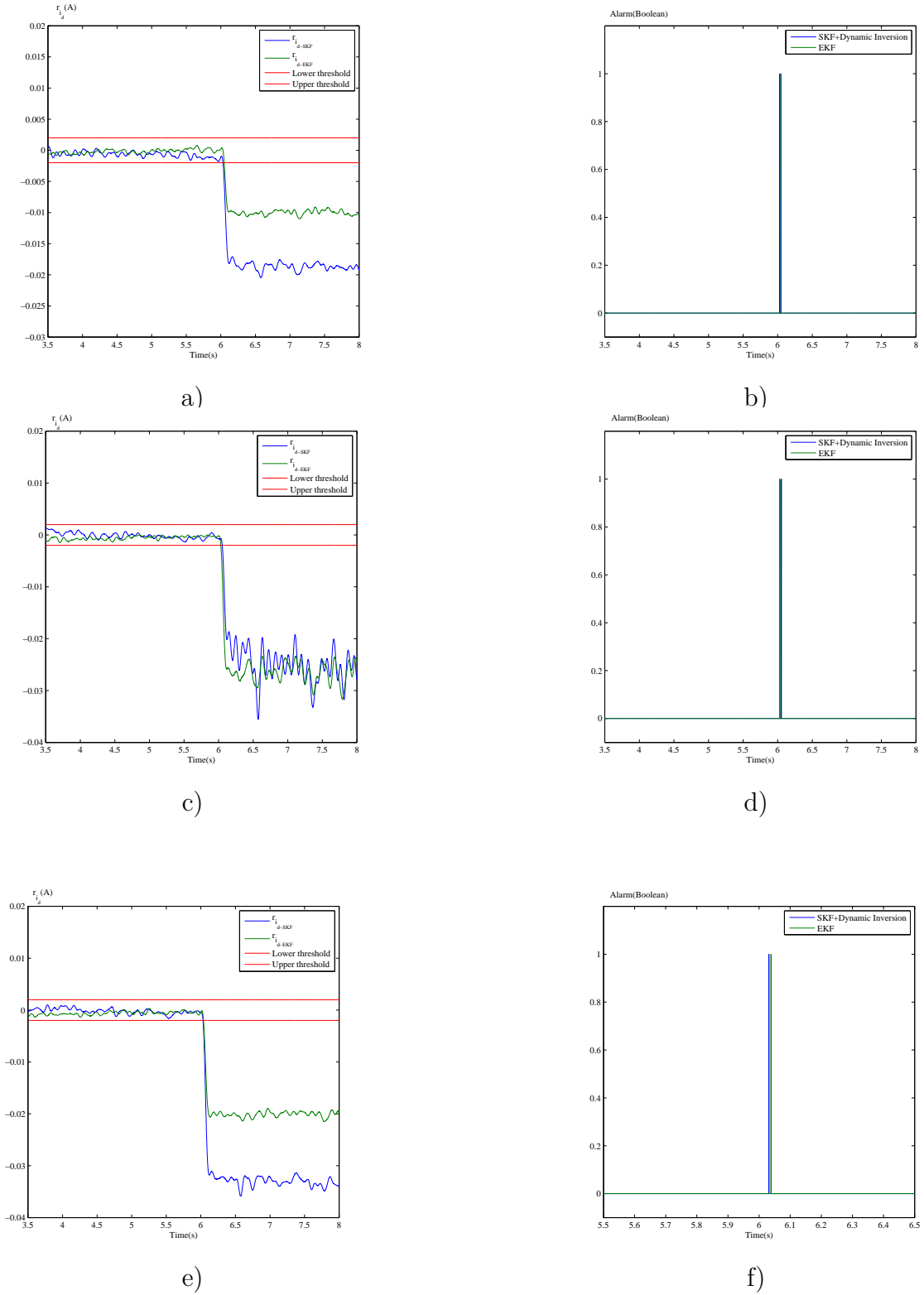


Figure 3.21: Residuals (a, c, e) and alarms (b, d, f) generation for 16%, 24%, 32% of shorted windings in one stator phase

Number of shorted windings (%)	EKF			SKF + Dynamic Inversion		
	False alarms	Response time (s)	Fault amplitude(A)	False alarms	Response time(s)	Fault amplitude(A)
8	YES	0,06	0,0025	NO	0,06	0,004
16	NO	0,05	0,01	NO	0,031	0,019
24	NO	0,032	0,025	NO	0,035	0,025
32	NO	0,038	0,025	NO	0,03	0,035

Figure 3.22: Diagnosis algorithms performance table

### 3.3.3 Conclusion

In this chapter the validation and integration of fault diagnosis algorithms on a test bench were realised where faults are shorted stator windings of a hybrid stepper motor (HSM).

In the first section, the control of the deteriorated (but functional ) HSM was accomplished by path planning on a simulation model embedded on a real-time machine. The wiring scheme of the bench was also presented.

In the second section, the validation and integration of diagnosis algorithms were realised. The linearisation of the HSM model by endogenous feedback presented in Chapter 1 was validated on the test bench. The inputs reconstruction of the equivalent linear system by dynamic inversion was also successfully completed. Next, a linear Kalman filter was used for residual generation. Fault detection was then realised by threshold crossing of the residual mean on a fault event. The same diagnosis approach was then realised with an extended Kalman filter (EKF), based on the nonlinear model of the HSM. False alarms, response time and residual amplitude change were compared for the two model based diagnosis methods. According to the test bench results of figure 3.22, the performances of dynamic inversion coupled to the SKF based diagnosis showed the following observations:

- There are no false alarms,
- The residual is fault sensitive,
- Low thresholds can be used,
- Input and states reconstruction is possible. In the case of system control purposes, only three sensors are required (currents and position sensors),

- There are no linearisation errors due to the linear equivalent model.

Regarding the EKF:

- False alarms occurred at low fault amplitudes,
- Fault-detection response time is slower than in the SKF based algorithm,
- Comparing to the SKF, the residual has a lower amplitude on a fault occurrence. Non-detection might result as a consequence

As described in Chapter 1, the linear model is equivalent at every value of the states paths, and not only at its equilibrium status. The linearisation process of the EKF is approximated and generates estimation errors [50] which might impact the robustness of the diagnosis. The inputs of the equivalent linear model were reconstructed with two methods, one based on endogenous feedback linearisation and the second one based on successive time derivatives of the flat outputs of the system. The last method is not suited for observer based diagnosis because the inputs are reconstructed with outputs measurements which are acquired after the fault event. As a result, the residual is not fault sensitive and no mean change is observable.





# Conclusion and perspectives

In the presented study, endogenous feedback linearisation was applied to a stepper motor of a flight critical actuator in order to improve model-based diagnosis algorithms. The proposed models were developed and tested on a test bench which was also designed and assembled. In the first Chapter, the fundamentals of differential flatness were presented and an example was given with the processing of a hybrid stepper motor's flat outputs. Different linearisation approaches were given such as endogenous feedback, pseudo-linearisation and tangent linearisation. The problem of local linearisation around an equilibrium point is solved for certain nonlinear systems with cancellation of their non-linearities by diffeomorphism and endogenous feedback. An example of linearisation of a stepper motor was shown in the first section. In the second section, model-based diagnosis tools such as residual generators, state estimators and observers for linear and nonlinear systems were shown.

In the second chapter, the design of a monitoring system for critical systems was described. First, safety critical systems were presented in the case of fuel systems and flight system control of different air planes and helicopters. In the first section, definitions and safety assessment methods provided by international airworthiness organisations were given. Also, a review on current system engineering tools and methods used in industry was developed in the second subsection of this chapter. In the second section, the case study of the safety assessment of a flight critical stepper motor was realized. The analytical models developed in Chapter 1 were proposed for a safe architecture based on analytical redundancy. The Development Assurance Level of the Architecture was also discussed. To conclude, analytic redundancy is a major advantage for safety critical architectures because analytic monitoring might lead to a decrease of the number of material redundancies actually present in current

flight critical architectures. On the other hand, difficulty of analytic redundancy designs relies in the exactitude of the model of the monitored system. The proposed architecture was designed with a safe and robust linearisation method reducing disturbances and fault-detection thresholds. As a result of this architecture, a patent dedicated to critical systems has been submitted.

In the third chapter, fault diagnosis algorithms were validated on a test bench where faults are shorted stator windings of a hybrid stepper motor (HSM).

In the first section, the control of the deteriorated (but functional ) HSM was accomplished by path planning on a simulation model embedded on a real-time machine. In the second section, the validation and integration of diagnosis algorithms were realised. The linearisation of the HSM model by endogenous feedback presented in chapter 1 was validated on the test bench. The inputs reconstruction of the equivalent linear system by dynamic inversion was also successfully completed. Next, a linear Kalman filter was used for residual generation. Fault detection was then realised by threshold crossing of the residual mean on a fault event. The same diagnosis approach was then realised with an extended Kalman filter (EKF), based on the nonlinear model of the HSM. False alarms, response time and residual amplitude change were compared for the two model-based diagnosis methods. As described in chapter 1, the linear model is equivalent at every value of the states paths, and not only at its equilibrium status. The linearisation process of the EKF is approximated and generates estimation which might impact the robustness of the diagnosis. The inputs of the equivalent linear model were reconstructed with two methods:

- the first one was based on endogenous feedback linearisation,
- and the second one was based on successive time derivatives of the flat outputs.

The second input reconstruction method is not suitable for diagnosis aims. Indeed, the reconstructed inputs are also affected by faults which does not allow the generation of a fault sensitive residual.

There are numerous perspectives resulting from this study. Indeed, health monitoring functions of critical systems need to be improved.

- The proposed algorithms can be extended with prognostic and ageing functions [103] in order to increase the systems health awareness,
- components lifetime can be optimised in order to reduce system maintenance operations,
- mechanism reconfiguration and recovery after fault detection can be coupled to the developed diagnosis algorithms for fault tolerance,
- the designed test bench could be improved by developing an electrical board, reducing the amount of wires generating measurement noise and voltage offsets,
- the developed diagnosis algorithms could be evaluated on a flight critical processor or FPGA regarding processing costs.

Also, a patent and international communications resulted from this work [12, 13, 11, 9, 10]. The patent is currently extending from Europe to an international application.



# Appendix A

## Differential geometry notions

### A.1 Differentiable manifold [73]

Given a differentiable mapping  $\Phi$  from  $\mathbb{R}^n$  to  $\mathbb{R}^{n-p}$  ( $0 \leq p < n$ ), we assume that there exists at least an  $x_0$  satisfying  $\Phi(x_0) = 0$  and that the tangent linear mapping  $D\Phi(x)$  has full rank  $(n - p)$  in a neighbourhood  $V$  of  $x_0$ . The set  $X$  defined by the implicit equation  $\Phi(x) = 0$ , is called differentiable manifold of dimension  $p$ . Otherwise stated:

$$X = \{x \in V | \Phi(x) = 0\} \quad (\text{A.1})$$

If in addition  $\Phi$  is  $k$ -times differentiable (respectively analytic),  $X$  is considered as a  $C^k$  differentiable manifold,  $k = 1, \dots, \infty$ .

**Example A.1.** *The sphere of  $\mathbb{R}^3$  centred at  $C$ , of coordinates  $(x_C, y_C, z_C)$ , and of radius  $R$ , given by  $\{(x, y, z) \in \mathbb{R}^3 | (x - x_C)^2 + (y - y_C)^2 + (z - z_C)^2 - R^2 = 0\}$ , is a 2-dimensional analytic manifold.*

### A.2 Diffeomorphism

Given two manifolds  $M$  and  $N$ , a differentiable mapping  $f : M \rightarrow N$  is called a diffeomorphism if it is a bijection and its inverse  $f^{-1} : N \rightarrow M$  is differentiable as well. If these functions are  $k$ -times continuously differentiable,  $f$  is called a  $C^k$ -diffeomorphism.

**Remark A.1.** *As a consequence to the invertibility property of a diffeomorphism, the Jacobian matrix of a diffeomorphism can not have its determinant equal to zero.*

## A.3 Implicit Function Theorem [73]

Let  $\Phi$  be a  $k$ -times continuously differentiable mapping, with  $k \geq 1$ , from an open set  $U \subset \mathbb{R}^n$  to  $\mathbb{R}^{n-p}$  with  $0 \leq p < n$ .

It is considered that there exists at least an  $x_0 \in U$  such that  $\Phi(x_0) = 0$ . If for every  $x$  in  $U$  the tangent linear mapping  $D\Phi(x)$  has full rank (equal to  $n - p$ ), there exists a neighbourhood  $V = V_1 \times V_2 \subset U$  of  $x_0$  in  $\mathbb{R}^n = \mathbb{R}^p \times \mathbb{R}^{n-p}$ , with  $V_1 \in \mathbb{R}^p$  and  $V_2 \in \mathbb{R}^{n-p}$ , and a  $k$ -times continuously differentiable mapping  $\Psi$  from  $V_1$  to  $V_2$  such that the two sets  $\{x \in V_1 \times V_2 | \Phi(x) = 0\}$  and  $\{(x_1, x_2) \in V_1 \times V_2 | x_2 = \Psi(x_1)\}$  are equal.

The function locally satisfies  $\Phi(x_1, \Psi(x_1)) = 0$  and the "dependent variable"  $x_2 = \Psi(x_1)$  is described by the  $p$  (locally) independent variables  $x_1$ .

## A.4 Module on a commutative ring

The notion of module is the natural generalisation of a vector space.

**Definition A.1.** *Considering  $A$ , a commutative ring. The  $A$ -module  $(M, +, \cdot)$  is a set defined by an internal law  $+$  and an external law  $A \times M \rightarrow M$ ,  $(\alpha, m) \mapsto \alpha m$  satisfying:*

- $(M, +)$  is an Abelian group.
- $\alpha(m + m') = \alpha m + \alpha m'$
- $(\alpha\beta)m = \alpha(\beta m)$
- $1.m = m$

for all  $\alpha, \beta \in A$  and all  $m, m' \in M$

## A.5 Particular matrices

**Definition A.2. (Hyper-regular matrices)** A matrix  $M \in \mathfrak{M}_{p,q} \left[ \frac{d}{dt} \right]$  is said hyper-regular if and only if its Smith decomposition leads to  $(I_p, 0_{p,q-p})$  if  $p < q$ , to  $I_p$  if  $p = q$ , and to  $\begin{pmatrix} I_q \\ 0_{p-q,q} \end{pmatrix}$  if  $p > q$ .

**Definition A.3. (Unimodular matrices)** A unimodular matrix  $M$  is a square integer matrix which determinant is equal to -1 or 1.

## A.6 Trivial Cartan field [73]

The trivial vector field on  $X \times \mathbb{R}_\infty^n$  is defined by:

$$\tau_X = \sum_{i \geq 0} \sum_{j=1}^n x_j^{(i+1)} \frac{\partial}{\partial x_j^{(i)}} \quad (\text{A.2})$$

Regarding the trivial vector field  $\tau_X$  corresponds the trivial system  $\dot{x}^{(j)} = x^{(j+1)}$  for all  $j$ , for which any infinitely differentiable function  $t \mapsto x(t)$  on  $X$  is an integral curve. Moreover,  $h$  being an arbitrary function, its Lie derivative along  $\tau_X$  is given by:

$$L_{\tau_X} h = \sum_{i \geq 0} \sum_{j=1}^n x_j^{(i+1)} \frac{\partial h}{\partial x_j^{(i)}} = \frac{dh}{dt} \quad (\text{A.3})$$

and  $\tau_X$  can thus be identified as the differential operator  $\frac{d}{dt}$ . The associated implicit system is empty, i.e. given by  $F \equiv 0$ .

## A.7 Tangent space and tangent bundle

Considering a given differentiable mapping  $\Phi$  from  $\mathbb{R}^n$  to  $\mathbb{R}^{n-p}$  ( $0 \leq p < n$ ), with at least an  $x_0$  satisfying  $\Phi(x_0) = 0$ . The tangent linear mapping  $D\Phi(x)$  of  $\Phi$  at  $x$ , expressed in the local coordinate system  $(x_1, \dots, x_n)$ , is thus the matrix  $\left( \frac{\partial \Phi_j}{\partial x_i}(x) \right)_{1 \leq i \leq n, 1 \leq j \leq n-p}$ . It is also assumed that  $D\Phi(x)$  has full rank  $(n - p)$  in

a neighbourhood  $V$  of  $x_0$ , so that the implicit equation  $\Phi(x) = 0$  defines a  $p$ -dimensional manifold denoted by  $X$ . It is verified that a normal vector at the point  $x$  to the manifold  $X$  is a linear combination of the rows of  $D\Phi(x)$ . Indeed, let  $y(t)$  be a differentiable curve contained in  $X$  for all  $t \in [0, \tau[$ , with  $\tau > 0$  sufficiently small, such that  $y(0) = x$ . Therefore  $\Phi(y(t)) = 0$  for all  $t \in [0, \tau[$  and thus

$$\frac{\Phi(y(t)) - \Phi(x)}{t} = 0. \quad (\text{A.4})$$

Letting  $t$  converge to 0, we get

$$D\Phi(x) \cdot \dot{y}(0) = 0, \quad (\text{A.5})$$

where

$$\dot{y}(0) \triangleq \frac{dy}{dt} \Big|_{t=0}, \quad (\text{A.6})$$

proving that the vector  $y(0)$ , tangent to  $X$  at the point  $x$ , belongs to the kernel of  $D\Phi(x)$  (Figure A.1). Doing the same for every curve contained in  $X$  and passing

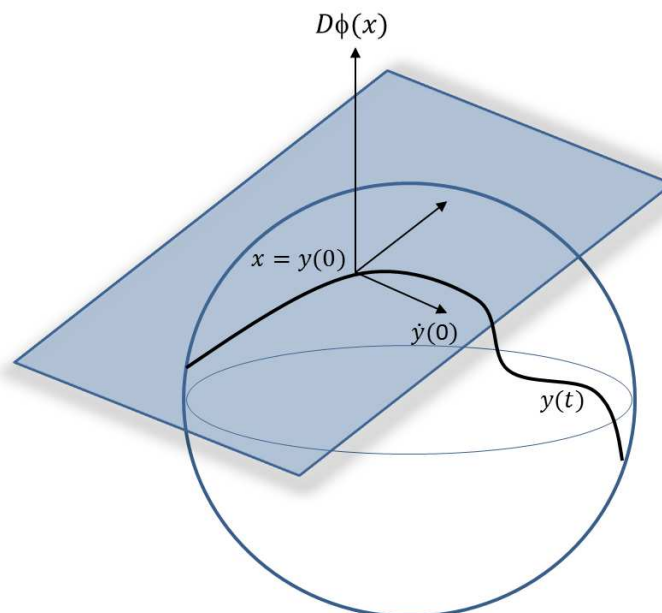


Figure A.1: Tangent and normal spaces to a manifold at a point

through  $x$ , it immediately results that every element of the range of  $D\Phi(x)$  is orthogonal to every tangent vector to  $X$  at the point  $x$ .



The *tangent space* to  $X$  at the point  $x \in X$  is the vector space  $T_x X = \ker D\Phi(x)$  and

the *tangent bundle*  $TX$  is the set

$$TX = \bigcup_{x \in X} T_x X \quad (\text{A.7})$$

.

Taking into account the fact that  $D\Phi(x)$  has rank  $n - p$  in  $V$ ,

$$\dim(T_x X) = \dim(\ker(D\Phi(x))) = p, \quad \forall x \in V \quad (\text{A.8})$$

.

## A.8 Differential form[73]

A *differential form of degree 1*, or *1-form* is defined by a  $C^\infty$ - section  $\omega$  of the cotangent bundle  $T^*X$ , *i.e.* a mapping for which, to each point  $x \in X$ , there corresponds an element  $\omega(x) \in T^*X$ ,  $\omega(x)$  being a linear combination of the local basis co-vectors of  $T^*X$  with  $C^\infty$  coefficients on  $X$ . The set of  $C^\infty$ -sections of  $T^*X$  is a vector space noted  $\Lambda^1(X)$ . The duality pairing between a 1-form

$$\omega = \sum_{i=1}^p \omega_i dx_i \quad (\text{A.9})$$

and a vector field

$$f = \sum_{i=1}^p f_i \frac{\partial}{\partial x_i} \quad (\text{A.10})$$

is given by

$$\langle w, f \rangle = \sum_{i=1}^p f_i \omega_i. \quad (\text{A.11})$$

A 1-form is not generally the differential of a function, as we now show, and consequently,  $\Lambda^1(X)$  contains more than the differentials of functions.



# Appendix B

## Stepper motor data-sheets

### B.1 Model parameters

Parameter	Description	Value	Unit
$R_0$	Stator winding resistance	8.7	$\Omega$
$Temp$	Ambient temperature	20	$^{\circ}\text{C}$
$L_i$	Inductance of the $i^{th}$ winding	$8.5 \cdot 10^{-3}$	H
$V_{bus}$	Supply voltage	24	V
$I_{max}$	Maximum current amplitude	0.55	A
$N_{tr}$	Number of teeth per rotoric wheel	50	N/A
$N_{sr}$	Number of steps per revolution	200	Steps/revolution
$N_s$	Number of stator teeth	40	N/A
$K_e$	<i>emf</i> constant	$7.3 \cdot 10^{-3}$	$V \cdot rad^{-1} \cdot s^{-1}$
$K_t$	Electromechanical torque constant	$7.3 \cdot 10^{-3}$	Nm/A
$\theta_a$	Mechanical step angle	1.8	Deg
$T_{df}$	Dry friction torque	$1 \cdot 10^{-9}$	Nm
$B$	Viscous friction torque of the motor	$5.7 \cdot 10^{-4}$	Nm

Parameter	Comment	Value	Unit
$n$	Electrical periods per revolution	50	N/A
$T_{hold}$	Holding torque	$8.7 \cdot 10^{-2}$	Nm
$T_d$	Detent torque	$8 \cdot 10^{-4}$	Nm
$T_{em}$	Electromechanical torque	$4 \cdot 10^{-3}$	Nm
$T_m$	Motor torque	$8.7 \cdot 10^{-2}$	Nm
$J_m$	Motor inertia (without load)	$1.1 \cdot 10^{-6}$	$Kgm^2$
$j_{mec}$	Load mounting backlash	$1 \cdot 10^{-5}$	Deg
$K_{LM}$	Stiffness ratio (load/motor)	$10 \cdot 0.15 \frac{180}{0.1\pi}$	Nm/rad
$J_L$	Load inertia	$3 \cdot 10^{-6}$	$Kgm^2$
$T_{rdf}$	Reference dry friction torque of the load	$6 \cdot 10^{-5}$	Nm
$T_{rvf}$	Load viscous friction torque	$2 \cdot 10^{-4}$	Nm
$p$	Number of poles pairs	50	N/A
$m$	Number of windings	2	N/A
$\omega_{max}$	Maximum angular velocity of the rotor-shaft	7	rad/s

Table B.1: Model parameters

# Appendix C

## Test bench specifications

### C.1 Hardware inputs and outputs

Functionality	Input(I) - Output(O)	Analog(A) - TTL - SPI	Signal levels	Nature of Analog Signals	Description
Absolute Shaft Encoder (SSI RS422 connexion)					
/Data	I	TTL	5V	voltage SE	PIN 1a Color grey
A+	I	TTL	5V	voltage SE	PIN 2a Color white/ green
0V Sensor	I	TTL	5V	voltage SE	PIN 3a Color red/ blue
B+	I	TTL	5V	voltage SE	PIN 4a Color green
/Clock	I	TTL	5V	voltage SE	PIN 5a Color green
DC 5V	O	A	5V DC	Encoder supply voltage	PIN 1b Color white
Clock	I	TTL	5V	voltage SE	PIN 2b Color yellow
B-	I	TTL	5V	voltage SE	PIN 3b Color grey/ pink
0V (Un) (GND)	I	A	GND		PIN 4b Color brown
A-	I	TTL	5V	voltage SE	PIN 5b Color brown/ green
Data	I	TTL	5V	voltage SE	PIN 6b Color pink
Dual Full Bridge PWM Motor Driver (Stepper Motor)					
PWM_A	O	TTL	5V	voltage SE	PWM signal n1 phase A
PWM_B	O	TTL	5V	voltage SE	PWM signal n2 phase A
PWM_C	O	TTL	5V	voltage SE	PWM signal n1 phase B
PWM_D	O	TTL	5V	voltage SE	PWM signal n2 phase B
/OTW	I	TTL	5V	voltage SE	Overtemperature warning signal
/FAULT	I	TTL	5V	voltage SE	Fault signal, open-drain, active-low
/RESET_AB	O	TTL	5V	voltage SE	Reset signal for half-bridge A and B
/RESET_CD	O	TTL	5V	voltage SE	Reset signal for half-bridge C and D
LEM Current Transducer (x2)					
M	I(x2)	A	±5V	voltage	PIN M (Vout)
Instrumentation Amplifier (x2)					
OUTPUT	I(x2)	A	±24V	voltage	voltage measurement
Torque sensor					
PIN C	I	A	±5V	voltage SE	+ signal
PIN D	I	A	0V	GND	- signal
PIN K	I	A	0V<u<30V	voltage SE	calibration control
Dual Full Bridge PWM Motor Driver (Dynamic Load)					
PWM_A	O	TTL	5V	voltage SE	PWM signal n1 phase A
PWM_B	O	TTL	5V	voltage SE	PWM signal n2 phase A
PWM_C	O	TTL	5V	voltage SE	PWM signal n1 phase B (in case of a 3 phased motor)
/OTW	I	TTL	5V	voltage SE	Overtemperature warning signal
/FAULT	I	TTL	5V	voltage SE	Fault signal, open-drain, active-low
/RESET_AB	O	TTL	5V	voltage SE	Reset signal for half-bridge A and B
/RESET_CD	O	TTL	5V	voltage SE	Reset signal for half-bridge C and D
Switch					
switch gate	O (x8)	TTL	5V	voltage SE	switch gate

Figure C.1: Sensors inputs and outputs

## C.2 Bench wiring diagram

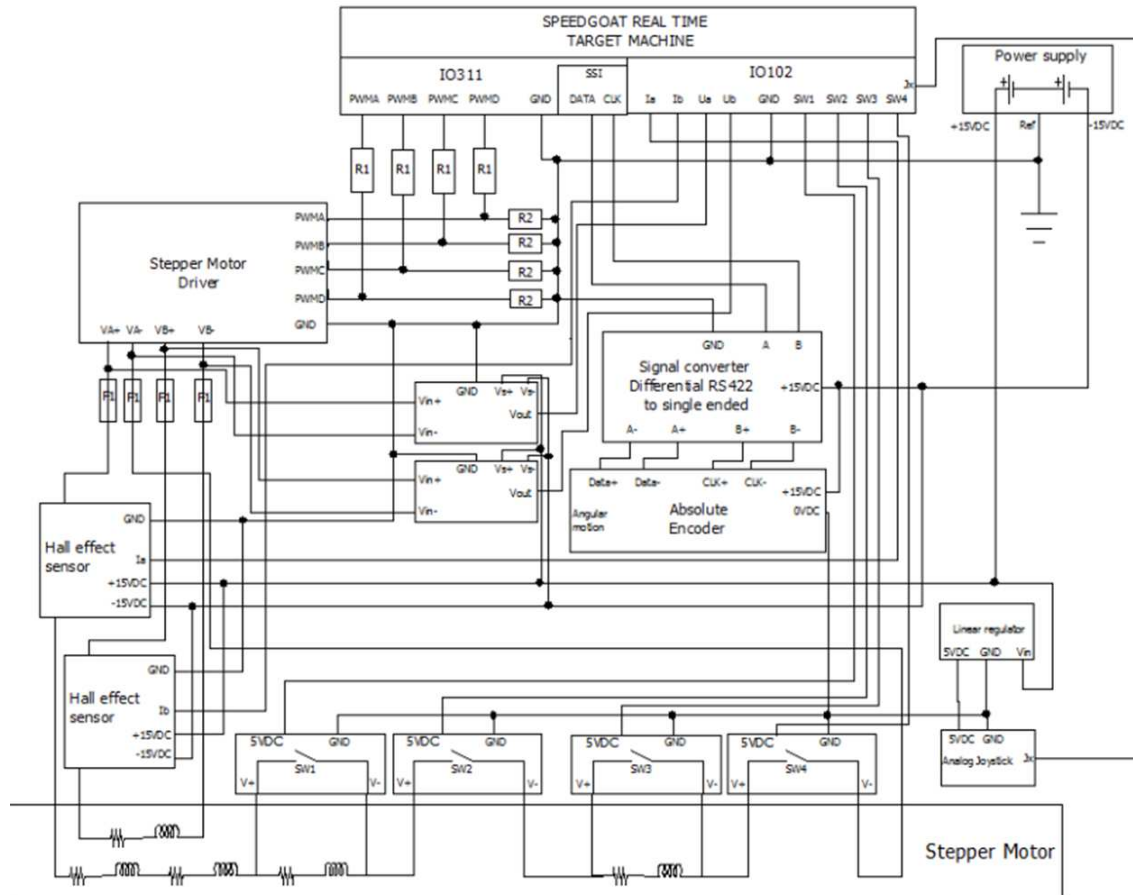


Figure C.2: Wiring diagram

## C.3 Power board specifications

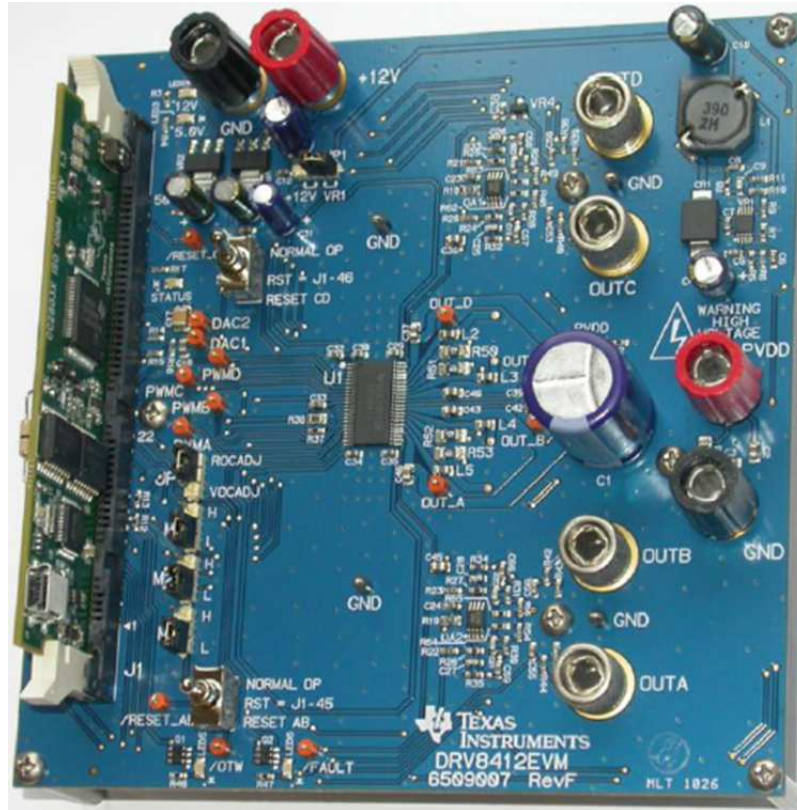


Figure C.3: Texas Instruments DRV8412 power board

The following features of the power board are required to control the bipolar hybrid stepper motor of the test bench:

- Dual H-Bridge Power Stage,
- 52 VDC maximum input voltage,
- 6A peak with a 3.5 A maximum continuous output current,
- Maximum of 500 KHz driver switching frequency,
- 12V control voltage can be supplied externally or regulated from the DC bus,
- Over current protection on the inverter stage.

## C.4 Real time machine specifications

In order to be able to interface the models on the host PC and the physical components of the test bench, the following specifications were required:

- an interface with the Matlab Simulink xPC Target environment. Speedgoat toolboxes and I/O modules are selectable from the xPC Target environment.
- an Intel Core i7 3.5 GHz CPU.

The inputs and outputs (I/O) connectivity were determined regarding the sensors and motor board connectivities and performances. The Solution proposal of the Speedgoat real time machine contained analog and digital I/O given by:

- 16 analog inputs, differential, 16-bit,  $\pm 10V$ ,  $\pm 5V$ ,  $\pm 2.5V$  (software selectable)
- 4 analog outputs, single-ended, 16-bit,  $\pm 10V$ ,  $\pm 5V$ ,  $\pm 2.5V$  (software selectable)
- 8 digital input, TTL
- 8 digital output, TTL
- 6 PWM generation, TTL. A PWM channel contains a 32-bit deep counter and the PWM control logic. The input clock for the counter and PWM control logic is 33MHz, which is sufficient compared to the 500 KHz PWM frequency limit of the stepper motor power board,
- 3 SSI Master, TTL. A differential to single ended adapter was required in order to connect the RS422 Absolute encoder, delivering a differential signal. The number of bits is software-tunable and limited to 32. A minimum of 13 bits is required by the absolute encoder.
- 34 digital input/output TTL.

The connexion between the host PC and the target machine is established with an Ethernet cable.



## C.5 Controllable switches: Reed relays

The following datasheets presents the specifications of the Reed relay used in the test bench.

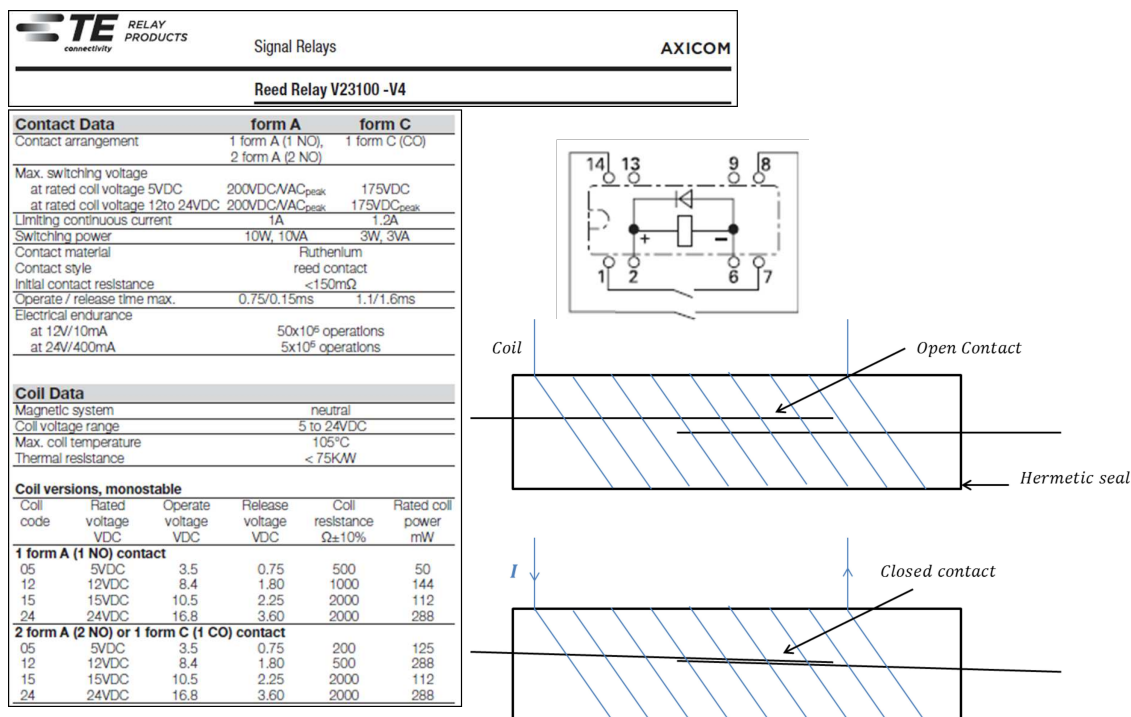


Figure C.4: TE Reed relay data sheet (source: <http://www.mouser.fr/TE-Connectivity/>)

When a current flows trough ports 2 and 6 of the wiring scheme depicted in figure C.4, the circuit is closed between ports 1 and 7 but also between ports 14 and 8. Otherwise, the circuits are open.



# References

- [1] ANSI/EIA 632. Processes for engineering a system. *American National Standards Institute/ Electronic Industries Alliance* (1999).
- [2] BARBOT, J.-P., BOUTAT, D., AND FLOQUET, T. An observation algorithm for nonlinear systems with unknown inputs. *Automatica* 45, 8 (2009).
- [3] BASSEVILLE, M., ABDELGHANI, M., AND BENVENISTE, A. Subspace-based fault detection algorithms for vibration monitoring. *Automatica* 36, 1 (2000), 101–109.
- [4] BEARD, R. V. *Failure accomodation in linear systems through self-reorganization*. PhD thesis, 1971.
- [5] BENDJEDIA, M. *Synthèse d’algorithmes de commande sans capteurs de moteurs pas à pas et implantation sur architecture programmable*. PhD thesis, 2007.
- [6] BERDJAG, D., CHRISTOPHE, C., COCQUEMPOT, V., AND JIANG, B. Non-linear model decomposition for robust fault detection and isolation using algebraic tools. *International Journal of Innovative Computing, Information & Control* 2, 6 (2006), 1337–1354.
- [7] BERGEON, B. *Commande robuste des systèmes automatisés*. Hermès Science Publications, Paris, 2001.
- [8] BERS, L. On hilbert’s 22<sup>nd</sup> problem. In *Mathematical Developments Arising From Hilbert Problems, Proceedings of Symposia in Pure Mathematics* (1976), pp. 559–609.

- 
- [9] BOBRINSKOY, A., CAZAURANG, F., BLUTEAU, B., GUERINEAU, O., AND GATTI, M. Model-based fault diagnosis for an electromechanical actuator of a helicopter turboshaft engine. *SAE 2014 Aerospace Systems and Technology Conference (ASTC)* (2014).
  - [10] BOBRINSKOY, A., CAZAURANG, F., GATTI, M., GUERINEAU, O., AND BLUTEAU, B. Model-based fault diagnosis of a flight-critical actuator. *Recent Advances in Aerospace Actuation Systems and Components* (2014), 84–89.
  - [11] BOBRINSKOY, A., GATTI, M., GUERINEAU, O., CAZAURANG, F., AND BLUTEAU, B. Dynamic inversion of a flight critical actuator for fault diagnosis. In *Digital Avionics Systems Conference (DASC), 2013 IEEE/AIAA 32nd* (Syracuse, New York), IEEE, pp. 1–16.
  - [12] BOBRINSKOY, A., GATTI, M., GUERINEAU, O., CAZAURANG, F., AND BLUTEAU, B. Model-based fault detection and isolation design for flight-critical actuators in a harsh environment. In *Digital Avionics Systems Conference (DASC), 2012 IEEE/AIAA 31st* (Williamsburg, Virginia), IEEE, pp. 1–17.
  - [13] BOBRINSKOY, A., GATTI, M., GUÉRINEAU, O., CAZAURANG, F., AND BLUTEAU, B. Système critique et procédé de surveillance. *Brevet FR1302661* (2013).
  - [14] BODSON, M., CHIASSE, J. N., NOVOTNAK, R. T., AND REKOWSKI, R. B. High-performance nonlinear feedback control of a permanent magnet stepper motor. *Control Systems Technology, IEEE Transactions on* 1, 1 (1993), 5–14.
  - [15] BOEHM, B. W. A spiral model of software development and enhancement. *Computer* 21, 5 (1988), 61–72.
  - [16] BOZZANO, M., AND VILLAFIORITA, A. *Design and safety assessment of critical systems*. CRC Press, 2010.

- [17] BYRNES, C. I., AND ISIDORI, A. *Asymptotic expansions, root-loci and the global stability of nonlinear feedback systems*. Springer, 1986, pp. 159–179.
- [18] CAZAURANG, F. *Commande robuste des systèmes plats Application à la commande d'une machine synchrone*. PhD thesis, Université Sciences et Technologies-Bordeaux I, 1997.
- [19] CHARLET, B. *Sur quelques problèmes de stabilisation robuste des systèmes non linéaires*. PhD thesis, 1989.
- [20] CHARLET, B., LÉVINE, J., AND MARINO, R. Sufficient conditions for dynamic state feedback linearization. *SIAM Journal on Control and Optimization* 29, 1 (1991), 38–57.
- [21] CHONG, S., WONG, C.-B., JIA, H., PAN, H., MOORE, P., KALAWSKY, R., AND O'BRIEN, J. Model driven system engineering for vehicle system utilizing model driven architecture approach and hardware-in-the-loop simulation. In *Mechatronics and Automation (ICMA), 2011 International Conference on*, IEEE, pp. 1451–1456.
- [22] CHRISTOPHE, C. *Surveillance des systèmes non linéaires: Application aux machines électriques*. PhD thesis, 2001.
- [23] DE LARMINAT, P. *Automatique: commande des systèmes linéaires*. Hermes Paris, 1993.
- [24] DE PERSIS, C., AND ISIDORI, A. A geometric approach to nonlinear fault detection and isolation. *Automatic Control, IEEE Transactions on* 46, 6 (2001), 853–865.
- [25] DESCUSSE, J., AND MOOG, C. H. Decoupling with dynamic compensation for strong invertible affine non-linear systems. *International journal of control* 42, 6 (1985), 1387–1398.
- [26] DING, S. X. *Model-based fault diagnosis techniques*, vol. 2013. Springer, 2008.

- 
- [27] DIOP, S., AND FLIESS, M. Nonlinear observability, identifiability, and persistent trajectories. In *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on* (1991), IEEE, pp. 714–719.
- [28] DRAKUNOV, S., AND UTKIN, V. Sliding mode observers. tutorial. In *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on* (1995), vol. 4, IEEE, pp. 3376–3378.
- [29] DUC, G, F. S. *Commande  $H_\infty$  et  $\mu$ -analyse, Des outils pour la robustesse*. HERMES Science Publications, Paris, 1999.
- [30] EDWARDS, C., LOMBAERTS, T., AND SMAILI, H. *Fault tolerant flight control: a benchmark challenge*, vol. 399. Springer, 2010.
- [31] ESTEFAN, J. A. Survey of model-based systems engineering (mbse) methodologies. *IncoSE MBSE Focus Group 25* (2007), 1–70.
- [32] EVENSEN, G. The ensemble kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics* 53, 4 (2003), 343–367.
- [33] FAIZ, N., AGRAWAL, S. K., AND MURRAY, R. M. Trajectory planning of differentially flat systems with dynamics and inequalities. *Journal of Guidance, Control, and Dynamics* 24, 2 (2001), 219–227.
- [34] FALB, P. L., AND WOLOVICH, W. Decoupling in the design and synthesis of multivariable control systems. *Automatic Control, IEEE Transactions on* 12, 6 (1967), 651–659.
- [35] FAR/CS-25. Acceptable means of compliance for large aeroplanes. *European Aviation Safety Agency (EASA) (former JAA)* (2008).
- [36] FILIPPOV, A. F. Differential equations with discontinuous right-hand side. *Matematicheskii sbornik* 93, 1 (1960), 99–128.
- [37] FLIESS, M. Some remarks on nonlinear invertibility and dynamic state feedback. *Theory and applications of nonlinear control systems* 8 (1986), 115–121.

- [38] FLIESS, M. Generalized controller canonical form for linear and nonlinear dynamics. *Automatic Control, IEEE Transactions on* 35, 9 (1990), 994–1001.
- [39] FLIESS, M., JOIN, C., AND MOUNIER, H. *An introduction to nonlinear fault diagnosis with an application to a congested internet router*. Springer, 2005, pp. 327–343.
- [40] FLIESS, M., JOIN, C., AND SIRA-RAMIREZ, H. Non-linear estimation is easy. *International Journal of Modelling, Identification and Control* 4, 1 (2008), 12–27.
- [41] FLIESS, M., LÉVINE, J., MARTIN, P., AND ROUCHON, P. Flatness and defect of non-linear systems: introductory theory and examples. *International journal of control* 61, 6 (1995), 1327–1361.
- [42] FLIESS, M., LÉVINE, J., MARTIN, P., AND ROUCHON, P. A lie-backlund approach to equivalence and flatness of nonlinear systems. *Automatic Control, IEEE Transactions on* 44, 5 (1999), 922–937.
- [43] FORSBERG, K., AND MOOZ, H. Application of the ?vee? to incremental and evolutionary development. *Systems Engineering in the Global Market Place* (1995), 801–808.
- [44] FORSBERG, K., AND MOOZ, H. The relationship of system engineering to the project cycle.
- [45] FOSSARD, A. J., AND NORMAND-CYROT, D. *Nonlinear Systems: Volume 1: Modeling and Estimation Volume 2: Stability and Stabilization Volume 3: Control*, vol. 2. Springer, 1996.
- [46] FRANCIS, B. A. *Lecture notes in control and information sciences*, 1987.
- [47] GAEID, K. S., AND MOHAMED, H. A. Diagnosis and fault tolerant control of the induction motors techniques a review. *Australian Journal of Basic and Applied Sciences* 4, 2 (2010), 227–246.

- 
- [48] GAUTHIER, J. P., AND BORNARD, G. Observability for any  $u(t)$  of a class of nonlinear systems. *Automatic Control, IEEE Transactions on* 26, 4 (1981), 922–926.
- [49] GLUMINEAU, A., AND MOOG, C. H. The essential orders and the non-linear decoupling problem. *International journal of control* 50, 6 (1989), 1825–1834.
- [50] GUSTAFSSON, F., AND GUSTAFSSON, F. *Adaptive filtering and change detection*, vol. 1. Wiley New York, 2000.
- [51] HAGENBLAD, A., GUSTAFSSON, F., AND KLEIN, I. A comparison of two methods for stochastic fault detection: the parity space approach and principal component analysis.
- [52] HERMANN, R., AND KRENER, A. J. Nonlinear controllability and observability. *IEEE Transactions on automatic control* 22, 5 (1977), 728–740.
- [53] HIRSCHORN, R. (a,b)-invariant distributions and disturbance decoupling of nonlinear systems. *SIAM Journal on Control and Optimization* 19, 1 (1981), 1–19.
- [54] HIRSCHORN, R. M. Invertibility of multivariable nonlinear control systems. *Automatic Control, IEEE Transactions on* 24, 6 (1979), 855–865.
- [55] HUNT, L., R, S., AND G, M. Design for multi-input nonlinear systems. *Differential geometric control theory* (1982), 268–298.
- [56] IEEE STD 1220-1998. Ieee standard for application and management of systems engineering process. *Institute for Electrical and Electronic Engineers* (1998).
- [57] ISERMANN, R. Process fault detection based on modeling and estimation methods a survey. *Automatica* 20, 4 (1984), 387–404.
- [58] ISERMANN, R. *Fault-diagnosis systems : an introduction from fault detection to fault tolerance*. Springer, Berlin, 2006.



- 
- [59] ISIDORI, A. *Nonlinear control systems*. Springer, Berlin, 1995.
- [60] ISIDORI, A., KRENER, A. J., GORI-GIORGI, C., AND MONACO, S. Non-linear decoupling via feedback: a differential geometric approach. *IEEE Transactions on automatic control* 26, 2 (1981), 331.
- [61] ISO/IEC 15288:2004. Systems engineering-system life cycle processes. *Institute for Electrical and Electronic Engineers* (2005).
- [62] JAKUBCZYK, B., AND NORMAND-CYROT, D. Automatique théorique. orbites de pseudo-groupes de difféomorphismes et commandabilité des systèmes non linéaires en temps discret. *Comptes rendus des séances de l'Académie des sciences. Série 1, Mathématique* 298, 11 (1984), 257–260.
- [63] JAKUBCZYK, B., AND RESPONDEK, W. On linearization of control systems. *Bull. Acad. Polonaise Sci. Ser. Sci. Math.* 28 (1980), 517–522.
- [64] JAMES, M. R. Controllability and observability of nonlinear systems.
- [65] JONES, H. L. *Failure detection in linear systems*. PhD thesis, 1973.
- [66] KADIMA, H. Méthodes et outils d'ingénierie des systèmes mécatroniques fiables. *EISTI* (2010), 6–15.
- [67] KRENER, A. J., AND RESPONDEK, W. Nonlinear observers with linearizable error dynamics. *SIAM Journal on Control and Optimization* 23, 2 (1985), 197–216.
- [68] KRITZINGER, D. *Aircraft system safety: Military and civil aeronautical applications*. Woodhead Publishing, 2006.
- [69] LAROCHE, E., SEDDA, E., AND DURIEU, C. Methodological insights for online estimation of induction motor parameters. *Control Systems Technology, IEEE Transactions on* 16, 5 (2008), 1021–1028.

- 
- [70] LAROCHE, E., SEDDA, E., DURIEU, C., AND LOUIS, J. Erreurs de modélisation d'une machine asynchrone-application au réglage d'un filtre de Kalman. *Revue Int. Génie Électrique* 3, 1 (2000), 7–37.
- [71] LAVIGNE, L., CAZAURANG, F., FADIGA, L., AND GOUPIL, P. New sequential probability ratio test: Validation on A380 flight data. *Control Engineering Practice* 22C (Oct. 2013), Pages 1–9.
- [72] LAVIGNE, L., ZOLGHADRI, A., GOUPIL, P., AND SIMON, P. A model-based technique for early and robust detection of oscillatory failure case in a380 actuators. *International Journal of Control, Automation and Systems (IJCAS)* 9, 1 (Feb. 2011), pp. 42–49.
- [73] LÉVINE, J. *Analysis and control of nonlinear systems: A flatness-based approach*. Springer, 2009.
- [74] LÉVINE, J. On necessary and sufficient conditions for differential flatness. *Applicable Algebra in Engineering, Communication and Computing* 22, 1 (2011), 47–90.
- [75] LÉVINE, J., AND NGUYEN, D. Flat output characterization for linear systems using polynomial matrices. *Systems & Control Letters* 48, 1 (2003), 69–75.
- [76] MACIEJOWSKI, J. M. Multivariable feedback design. *Electronic Systems Engineering Series, Wokingham, England: Addison-Wesley, c1989* 1 (1989).
- [77] MAHAMOUD, A., GLUMINEAU, A., AND SOULEIMAN, I. FDI using High Gain Observers for Cascade Systems: application to induction motors. *European Control Conference ECC'09* (23-26 August 2009).
- [78] MAHAMOUD, A., GLUMINEAU, A., AND SOULEIMAN, I. Methodology for nonlinear FDI Observer via Nonlinear Transformation: Application to a DC Serie Motor. In *Proceeding IFAC SAFEPROCESS'09* (June-July 2009), pp. 30–35.

- 
- [79] MARINO, R. On the largest feedback linearizable subsystem. *Systems & Control Letters* 6, 5 (1986), 345–351.
- [80] MARTIN, P. *Contribution à l'étude des systèmes différentiellement plats*. PhD thesis, 1992.
- [81] MARTINEZ-GUERRA, R., AND MATA-MACHUCA, J. L. Fault detection and diagnosis in nonlinear systems. *AMC* 10 (2014), 12.
- [82] MARTINEZ TORRES, C., LAVIGNE, L., CAZAURANG, F., ALCORTA GARCIA, E., AND DIAZ, D. Flatness-based fault tolerant control. *DYNA* 81, 188 (Dec. 2014), 130–137.
- [83] MARTINEZ TORRES, C., LAVIGNE, L., CAZAURANG, F., ALCORTA GARCIA, E., AND DIAZ, D. Flatness-based fault tolerant control. *DYNA* 81, 188 (Dec. 2014), 130–137.
- [84] MASSOUMNIA, M.-A. A geometric approach to the synthesis of failure detection filters. *Automatic Control, IEEE Transactions on* 31, 9 (1986), 839–846.
- [85] MILAM, M. B., FRANZ, R., HAUSER, J. E., AND MURRAY, R. M. Receding horizon control of vectored thrust flight experiment. *IEE Proceedings-Control Theory and Applications* 152, 3 (2005), 340–348.
- [86] MULTON, B. Les machines synchrones autopilotées. *Préparation à l'agrégation de génie électrique, ENS de Cachan* (2004).
- [87] NIJMEIJER, H., AND RESPONDEK, W. Dynamic input-output decoupling of nonlinear control systems. *Automatic Control, IEEE Transactions on* 33, 11 (1988), 1065–1070.
- [88] NSWC-98/LE1. Handbook of reliability prediction procedures for mechanical equipment. *Naval Surface Warfare Center* (1998).
- [89] QUADRAT, A., AND ROBERTZ, D. Computation of bases of free modules over the weyl algebras. *Journal of Symbolic Computation* 42, 11 (2007), 1113–1141.

- [90] ROYCE, W. W. Managing the development of large software systems. In *proceedings of IEEE WESCON* (1970), vol. 26, Los Angeles.
- [91] RTCA DO-160. *Environmental Conditions and Test Procedures for Airborne Equipment, Radio Technical Commission for Aeronautics, issue F 6* (2007).
- [92] RTCA DO-178B/ED12 . *Software considerations in airborne systems and equipment certification* (1992).
- [93] RTCA DO-254/ED80. Design assurance guidance for airborne electronic hardware.
- [94] SAE ARP 4754/ED79. Certification considerations for highly-integrated or complex aircraft systems. *SAE, Warrendale, PA* (1996).
- [95] SAE ARP 4761. Guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment. *SAE, Warrendale, PA* (1996).
- [96] SIMON, P., LAVIGNE, L., ZOLGHADRI, A., AND GOUPIL, P. Détection de pannes oscillatoires dans une chaîne d’asservissement en position d’une gouverne de l’A380. In *CIFA 2008* (Bucarest, Romania, Sept. 2008), p. ...
- [97] SINGH, S. N. Decoupling of invertible nonlinear systems with state feedback and precompensation. *IEEE Transactions on automatic control*, 6 (1980), 1237.
- [98] SINGH, S. N. A modified algorithm for invertibility in nonlinear system. *IEEE Transactions on automatic control*, 2 (1981), 595.
- [99] SLOTINE, J.-J., HEDRICK, J., AND MISAWA, E. Nonlinear state estimation using sliding observers. In *Decision and Control, 1986 25th IEEE Conference on* (1986), vol. 25, IEEE, pp. 332–339.

- [100] UTKIN, V. I. Sliding modes in multidimensional systems with variable structure. In *Decision and Control including the 12th Symposium on Adaptive Processes, 1973 IEEE Conference on* (1973), pp. 727–727.
- [101] VAN NIEUWSTADT, M. J., AND MURRAY, R. M. Real time trajectory generation for differentially flat systems.
- [102] VENKATASUBRAMANIAN, V., RENGASWAMY, R., YIN, K., AND KAVURI, S. N. A review of process fault detection and diagnosis: Part 1: Quantitative model-based methods. *Computers and chemical engineering* 27, 3 (2003), 293–311.
- [103] VINSON, G., RIBOT, P., PRADO, T., AND COMBACAU, M. A generic diagnosis and prognosis framework: application to permanent magnets synchronous machines. In *Prognostics and System Health Management Conference*, pp. 1039–1044.
- [104] VON LÖWIS, J., AND RUDOLPH, J. *Real-time trajectory generation for flat systems with constraints*. Springer, 2003, pp. 385–394.
- [105] WONHAM, W. M. *Linear multivariable control*. Springer, 1974.
- [106] WONHAM, W. M., AND MORSE, A. S. Decoupling and pole assignment in linear multivariable systems: a geometric approach. *SIAM Journal on Control* 8, 1 (1970), 1–18.
- [107] ZHOU, K., DOYLE, J. C., AND GLOVER, K. *Robust and optimal control*, vol. 40. Prentice Hall New Jersey, 1996.
- [108] ZWINGELSTEIN, G. *Diagnostic des défaillances: Théorie et pratique pour les systèmes industriels*. Hermès, 1995.

## Algorithmes et architectures pour la commande et le diagnostic des systèmes critiques de vol

**Résumé :** Les systèmes critiques de vol tels que les actionneurs électromécaniques ainsi que les calculateurs de commande moteur (ECU) et de vol (FCU), sont conçus en tenant compte des contraintes aéronautiques sévères de sûreté de fonctionnement. Dans le cadre de cette étude, une architecture calculateur pour la commande et la surveillance d'actionneurs moteur et de surfaces de vol est proposée et a fait l'objet d'un brevet [13]. Pour garantir ces mesure de sûreté, les ECU et FCU présentent des redondances matérielles multiples, mais engendrent une augmentation de l'encombrement, du poids et de l'énergie consommée. Pour ces raisons, les redondances à base de modèles dynamiques, présentent un atout majeur pour les calculateurs car elles permettent dans certains cas de maintenir les exigences d'intégrité et de disponibilité tout en réduisant le nombre de capteurs ou d'actionneurs. Un rappel sur les méthodes de diagnostic par générateurs de résidus et estimateurs d'états [58, 26, 47] est effectué dans cette étude. Les propriétés de platitude différentielle et la linéarisation par difféomorphisme et bouclage endogène [80, 41, 73] permettent d'utiliser des modèles linéaires équivalents avec les générateurs de résidus. Un banc d'essai a été conçu afin de valider les performances des algorithmes de diagnostic.

**Mots-clés :** diagnostic, inversion dynamique, platitude différentielle, systèmes critiques de vol.

## Algorithms and Architectures for Control and Diagnosis of Flight Critical Systems

**Abstract :** Flight-Critical Systems such as Electromechanical Actuators driven by Engine Control Units (ECU) or Flight Control Units (FCU) are designed and developed regarding drastic safety requirements. In this study, an actuator control and monitoring ECU architecture based on analytic redundancy is proposed. In case of fault occurrences, material redundancies in avionic equipment allow certain critical systems to reconfigure or to switch into a safe mode. However, material redundancies increase aircraft equipment size, weight and power (SWaP). Monitoring based on dynamical models is an interesting way to further enhance safety and availability without increasing the number of redundant items. Model-based fault detection and isolation (FDI) methods [58, 26, 47] such as observers and parity space are recalled in this study. The properties of differential flatness for nonlinear systems [80, 41, 73] and endogenous feedback linearisation are used with nonlinear diagnosis models. Linear and nonlinear observers are then compared with an application on hybrid stepper motor (HSM). A testing bench was specially designed to observe in real-time the behaviour of the diagnosis models when faults occur on the stator windings of a HSM.

**Keywords :** diagnosis, dynamic inversion, differential flatness, flight critical systems.