

Bond Graph Model Based on Structural Diagnosability and Recoverability Analysis : Application to Intelligent Autonomous Vehicles

THÈSE

présentée et soutenue publiquement le 6 Décembre 2012

pour l'obtention du

Doctorat de l'université Lille 1

(spécialité Automatique et Informatique Industrielle)

par

Rui LOUREIRO

Composition du jury

<i>Président :</i>	Mohamed DJEMAI	Professeur à Université de Valenciennes
<i>Rapporteurs :</i>	Eric BIDEAUX Taha BOUKHOBZA	Professeur à INSA Lyon Professeur à Université de Lorraine
<i>Examineur :</i>	François CHARPILLET	Directeur de recherche INRIA Nancy
<i>Invité :</i>	Geneviève DAUPHIN-TANGUY	Professeur à L'école Centrale de Lille
<i>Directeurs de thèse :</i>	Belkacem OULD BOUAMAMA Rochdi MERZOUKI	Professeur à l'université de Lille1 Professeur à l'université de Lille1

To my mother Isabel,
To my brother João,
To my Grandparents Lucinda and Fernando.

Acknowledgements

It is time to write the acknowledgments. I will do my best.

First of all, I would like to show my gratitude towards my supervisors, Mr. Belkacem Ould-Bouamama and Mr. Rochdi Merzouki for providing me with the opportunity to work on this interesting topic. Without whose support, availability, encouragement and fruitful discussions, I would not have achieved the results I obtained in my Ph.D. thesis. It was through his persistence, understanding and kindness that I always worked with enthusiasm.

My gratitude also goes towards Mr. Taha Boukhobza and Mr. Eric Bideaux for their availability and interest on reviewing my work. I express to them my deepest appreciation. I also thank Ms. Geneviève Dauphin-Tanguy, Mr. François Charpillat and Mr. Mohamed Djemai for agreeing to be part of the jury of this thesis.

Thank you to doctoral and post-doctoral researcher at the Laboratoire d'Automatique, Génie Informatique et Signal (LAGIS). With them, I was able to share the joys and difficulties faced during a Ph.D. thesis. More specifically, I would like to extend very special thanks my office colleagues Samir Benmoussa and Youcef Touati for the helpful discussions and their useful suggestions during these three years.

Thank you to all persons who were not mentioned above, but were involved with our work, directly and indirectly.

Finally, thanks to my family for the support they provided me through my entire life. I must acknowledge Alice, without whose encouragement and assistance, I would not have worked serenely.

Contents

List of Figures	vii
List of Tables	xi
Résumé	xiii
Introduction	1
1 State of the Art	11
1.1 Introduction	12
1.2 Fault tolerant systems	12
1.2.1 Fault diagnosis	14
1.2.2 Active Fault tolerant control	19
1.3 Control reconfigurability analysis and problem statement	25
1.3.1 Control reconfigurability analysis	25
1.3.2 Problem statement	32
1.4 Why use the bond graph tool for fault recoverability analysis?	34
1.5 Conclusions	35
2 Structural recoverability analysis from a bond graph model	37
2.1 Introduction	38
2.2 Interest of structural analysis	38
2.3 The fault tolerant control problem	39
2.4 Bond graph model for structural fault diagnosis analysis	41
2.4.1 Bond graph representation	41
2.4.2 Causality and bicausality on the BG model	44
2.4.2.1 Causality	44
2.4.2.2 Bicausality	45
2.4.3 Direct Current motor (DC-motor) example	46

CONTENTS

2.4.4	Generation of <i>ARRs</i> from BG model	47
2.4.5	Fault isolation	52
2.4.6	Generation of additional non-redundant <i>ARRs</i>	53
2.5	Bond graph for structural recoverability analysis	58
2.6	Conclusions	70
3	Local adaptive fault compensation	73
3.1	Introduction	74
3.2	Fault modeling	75
3.3	Model inversion	83
3.4	Bond graph fault tolerant control through power variable compensation	87
3.5	Conclusions	90
4	Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle	93
4.1	Introduction	94
4.2	Intelligent autonomous vehicle description and modeling	94
4.2.1	RobuTAINeRs quarter vehicle modeling	96
4.3	Structural analysis for fault tolerance	103
4.3.1	Diagnosis procedure	103
4.3.2	Structural fault tolerance analysis procedure	106
4.3.3	Structural analysis for local adaptive compensation	112
4.4	Conclusions	117
	Conclusions and Recommendations for future research	119
A	Bond Graph generalities	123
A.1	Bond graph modeling	123
A.2	Causality	128
A.3	State space equations associated to a bond graph model	130
A.4	Bicausal assignment procedure	131
A.5	Bond graph LFT for fault estimation	132
A.5.1	Linear fractional transformation (BG-LFT) for fault modeling .	132
A.5.1.1	Input fault modeling	136
A.5.1.2	Sensor fault modeling	137
A.5.2	Methodology for fault estimation	138
A.5.2.1	Fault parameter estimation	139

A.5.2.2 Input fault estimation 140

A.5.2.3 Sensor fault estimation 141

Bibliography **143**

CONTENTS

List of Figures

1	Organization of the works developed within the group MOCIS.	3
2	Block diagram representation of a system with direct sensor redundancy.	6
1.1	General architecture of an AFTC system.	13
1.2	Classification of diagnosis methods.	16
1.3	Quantitative Model-based fault detection and isolation (FDI) based on residuals.	19
1.4	Fault accommodation scheme.	20
1.5	System reconfiguration scheme.	21
1.6	General scheme of fault recoverability analysis found in literature.	33
1.7	Proposed scheme of fault recoverability analysis. In this case, the real diagnosis information is considered, together with plant, actuator and sensor faults.	33
2.1	General representation of a bond graph model.	42
2.2	Causal BG of the connection between two physical subsystem (S_1, S_2) and corresponding block diagram representation.	44
2.3	Bicausal BG of the connection between two physical subsystem (S_1, S_2) and corresponding block diagram representation.	45
2.4	Schematic of a DC-motor.	46
2.5	WBG and BG model of the DC-motor in integral causality.	47
2.6	(a,c) detector of effort (De), and of flow (Df) and (b,d) signal source of effort (SSe), and of flow (SSf), respectively.	49
2.7	Direct detector redundancy in a BG model for diagnosis in preferred derivative causality: (a) Violation of the causality assignment rules if both $SSf : y_1$, and $SSe : y_2$ are dualized, (b) only $SSf : y_1$ is dualized (no violation).	50
2.8	BG model of the DC-motor in derivative causality with dualized detectors.	51

LIST OF FIGURES

2.9	General representation of a Bond graph model for diagnosis.	54
2.10	General representation of a Bond graph model for diagnosis without dualizing $Df : y_1$	54
2.11	General representation of a Bond graph model for diagnosis in bicausality.	56
2.12	Bond graph model of DC-motor in bicausality for angular velocity estimation $\hat{\omega}$	56
2.13	General representation of the BG (a) healthy system (b) Faulty system with removed detector outer vertex (faulty sensor).	59
2.14	General representation of the BG (a) healthy system (b) faulty system with removed actuator outer vertex (faulty actuator).	62
2.15	Material input redundancy: Direct causal path from effort sources (S_e), part (a) (flow sources (S_f), part (b)) to sensor involving the same elements.	63
2.16	Some controlled BG elements that can cause a PB: part (a) Controlled gyrator ($MGY : k$), (b) Controlled transformer ($MTF : m$), and (c) Controlled resistive element ($R : R_1$).	66
2.17	Algorithm of structural fault recoverability analysis.	69
3.1	Comparison between the system outputs and its forward model in fault free case (a), and in faulty situation of the system (b).	76
3.2	General acausal BG model.	77
3.3	Acausal BG model with an additional source imposing the power generated by the fault. In part (a) <i>1-junction</i> , the power is represented as a modulated source of effort $MSe : -e_F$, and in part (b) <i>0-junction</i> , the power is represented as a modulated source of flow $MSf : -f_F$	78
3.4	General BG model for diagnosis with healthy sensor (a) and with faulty sensor (b).	79
3.5	BG model of the DC-motor in preferred integral causality, with a fault in the GY element.	80
3.6	BG model of the DC-motor in bicausality for estimation of F_{GY}	81
3.7	General representation of a BG direct model.	84
3.8	General representation of a BG inverse model.	84
3.9	I/O power line, deduced from the acausal BG model of the DC-motor.	86
3.10	Inverse BG model of the DC-motor with the representation in the form of block diagram of the DC-motor physical process.	87
3.11	BG inverse model for GY fault compensation.	88

LIST OF FIGURES

3.12	(a) - R -element in conductive causality associated to a 1-junction, (b) - BF-LFT for fault modeling of the R -element in conductive causality.	89
3.13	(a) - R -element in resistive causality associated to a bicaused 1-junction, (b) - BF-LFT for fault modeling of the R -element in resistive causality.	89
3.14	Local adaptive compensation schme in the form of block diagram. . . .	91
3.15	Simulation results when the fault is not compensated.	91
3.16	Simulation results when the fault is compensated.	92
4.1	Confined space of port terminal.	95
4.2	(a) CAD figure of RobuTAINeRs prototype, (b) Real RobuTAINeR picture.	95
4.3	CAD figure of RobuTAINeRs wheel frame with steering and traction motor.	96
4.4	Word bond graph of a quarter RobuTAINeRs (electromechanical subsystem).	97
4.5	Bond graph of a quarter RobuTAINeRs (electromechanical subsystem).	98
4.6	BG model of RobuTAINeR's.	102
4.7	BG model of the j^{th} electromechanical subsystem in derivative causality.	104
4.8	BG model of the j^{th} electromechanical subsystem in bicausality for estimation of current \hat{i}_j	105
4.9	BG of the 1 st electromechanical system with a PB label in the $MGY : k_{e1}$ element.	109
4.10	<i>SCANeR Studio</i> driving simulator and the wheel frame of RobuTAINeR that was modeled with <i>Matlab/Simulink</i>	111
4.11	Radicatel terminal with desired trajectory on SCANeR Studio software.	112
4.12	Residuals of the front left electromechanical subsystem, under input faulty conditions	112
4.13	Desired trajectory (solid line) and tracking results when operating under input faulty (dotted line) conditions.	113
4.14	(a) BG-LFT for fault estimation, and (b) inverse BG-LFT for system inversion.	115
4.15	Desired velocity of RobuTAINeR's wheels and required voltage obtained from system inversion.	116
4.16	Residuals 1, 2, and 3 of the front left electromechanical subsystem. . . .	116
4.17	Residuals 4, and 5 of the front left electromechanical subsystem. . . .	117
4.18	Fault estimation, tracked angular velocity, and tracking error.	117

LIST OF FIGURES

A.1	Active elements: Sources of effort and flow (a) , modulated sources of effort and flow (b)	124
A.2	Representation of 1- port passive elements: R -element (a) , I -element (b) , and C -element.	124
A.3	2-port TF -element.	126
A.4	2-port GY -element.	126
A.5	BG θ -junction.	126
A.6	BG 1 -junction.	127
A.7	(a) , (b) representation of a detector of effort and flow, respectively.	128
A.8	(a) Storage elements I and C in integral causality, (b) Storage elements I and C in derivative causality.	129
A.9	(a) bicausal assignment in a θ -junction, (b) bicausal assignment in a 1 -junction.	131
A.10	Block diagram LFT of an R -element in resistive causality with a multiplicative fault.	132
A.11	BG model LFT of an R -element in resistive causality with a multiplicative fault.	133
A.12	BG-LFT for parameter fault modeling.	134
A.13	BG-LFT for R -parameter fault modeling in conductive causality.	135
A.14	BG-LFT for fault modeling on GY elements.	136
A.15	BG-LFT for fault modeling on TF elements.	136
A.16	BG-LFT for actuator fault modeling	137
A.17	BG-LFT for sensor fault modeling: (a) sensor of flow, (b) sensor of effort	138
A.18	(a) BG-LFT for R -parameter fault modeling, (b) Bicausal BG-LFT for R -parameters fault estimation.	139
A.19	(a) BG-LFT for MSE fault modeling, (b) Bicausal BG-LFT for MSE fault estimation.	140
A.20	(a) BG-LFT for Df fault modeling, (b) Bicausal BG-LFT for Df fault estimation.	141

List of Tables

1.1	Comparison of different procedures for control reconfigurability analysis.	31
2.1	Variables of the simplified DC-motor system.	46
2.2	Fault signature matrix (FSM).	52
2.3	Fault Signature matrix of the DC-motor from bond graph.	53
2.4	Fault Signature matrix of the DC-motor from BG with the additional non-redundant residuals.	57
2.5	Fault Signature and Recoverability Matrix (FSRM).	69
4.1	Fault Signature Matrix (FSM) of the j^{th} electromechanical system. . . .	105
4.2	Fault Signature Matrix (FSM) of the j^{th} electromechanical system. . . .	107
4.3	Fault Signature and recoverability Matrix (FSRM) of the j^{th} electromechanical system.	110
4.4	System parameters	113
A.1	Effort and flow variables in different physical domains	123
A.2	Causality of BG elements.	130

LIST OF TABLES

Résumé

Au cours de ces dernières années, la présence de systèmes de commandes dans plusieurs systèmes industriels est devenue essentielle voire même primordiale. Et cela, afin de permettre aux systèmes industriels d'atteindre des performances nécessaires aux termes de rendements et une adaptabilité aux différents champs d'applications. Un système de commande est basé sur le développement des lois de commandes. Elles sont conçues pour atteindre des objectifs en supposant que le système est en fonctionnement normal. Cependant, les systèmes automatiques sont généralement assujettis aux perturbations, défauts, pannes. Par exemple, l'apparition d'un défaut au sein d'un système peut engendrer son dysfonctionnement. Si le défaut est jugé critique pour le système (une centrale nucléaire, un avion), ce dernier peut causer des accidents catastrophiques. Par conséquent, le concept de systèmes tolérants aux défauts (Fault tolerant systems) est introduit et a pour objectif d'assurer la sécurité et la prévention des accidents dans ces systèmes critiques pour la sécurité des personnes et de l'environnement. Ce concept a fait l'objet de plusieurs travaux de recherche à savoir le diagnostic de défauts (Fault Diagnosis, FD) et la commande tolérante aux défauts (Fault Tolerant Control, FTC).

Le diagnostic de défauts consiste à développer des algorithmes permettant de détecter, d'isoler et d'identifier le défaut une fois ce dernier apparaît sur le système. Quant à la commande tolérante aux défauts, elle consiste à reconcevoir les lois de commandes en tenant compte le type et l'amplitude de défaut et cela afin de permettre au système défectueux d'atteindre ces objectifs initiaux tout en maintenant la sûreté globale de ce dernier. Le couplage entre FD et FTC est pleinement intégré dans la conception de système dynamique sûr de fonctionnement. Et on peut trouver plusieurs champs d'application : automobile, production, système robotiques, etc.

L'objectif principal des algorithmes tolérants aux défauts:

Definition 0.1. ... FTC est un système de commande qui possède la capacité de compenser automatiquement l'effet de certains défauts affectant les composants du système en maintenant la stabilité globale de ce dernier [Zhang 2008].

En ce qui concerne la conception des systèmes tolérants aux défauts, il existe deux catégories [Patton 1997]: commande tolérante aux défauts passives (Passive FTC) et commande tolérante aux défauts active (Active FTC). La commande tolérante aux défauts passive utilise des contrôleurs robustes à structure fixe en considérant un ensemble de défaillances probables ce qui pose une limite de ces approches [Hsieh 2002, Liao 2002, Niemann 2005]. Par contre, la commande tolérante aux défauts active réagit en ligne dès l'apparition du défaut [Mhaskar 2006, Zhang 2009, Miksch 2008].

Des différentes approches ont été développées pour la conception et la mise en oeuvre des procédures de diagnostic. Ces approches dépendent de la nature de la connaissance utilisée pour décrire le fonctionnement du système. Elles peuvent être classifiées en deux groupes: les approches à base de modèle [Frank 1990, Frank 1997a, Gertler 1997], et les approches à base de l'analyse des données [Li 2000, Srinivas 1994, Qian 2008].

Concevoir un système tolérant aux défauts n'est pas de la tâche facile, puisqu'il faut considérer à la fois les informations issues de l'étape du diagnostic et les stratégies de la commande tolérante aux défauts. Aussi, un système peut être tolérant aux défauts si et seulement s'il existe une redondance aux termes d'information matérielle ou analytique.

La redondance matérielle consiste à remplacer un composant du système, par exemple un actionneur, par un composant ayant les mêmes caractéristiques techniques. Les principaux inconvénients de la redondance matérielle sont généralement liés aux coûts de fabrication, d'exploitation et de maintenance. De même, l'installation de nouveaux équipements supplémentaires sur le système n'est pas toujours possible vu le manque d'un espace supplémentaire nécessaire pour le système.

Quant à la redondance analytique, elle consiste à identifier des relations mathématiques permettant de calculer (ou estimer) d'une manière analytique la fonction perdue à cause du défaut et d'utiliser cette information pour planifier une stratégie de reconfiguration en utilisant les composants sains du système. Cette tâche est réalisée en utilisant les relations statiques / dynamiques entre les variables du système [Wu 2000]. Par exemple, un système présente une redondance analytique si quand un actionneur tombe en

panne, le système de commande peut assurer la commande globale du système en utilisant les autres actionneurs [Kwong 1995]. La redondance analytique est aussi utilisée lors de l'étape de diagnostic pour comparer le comportement réel du système avec un comportement de référence représentant le système dans des conditions nominales.

Les travaux existant sur la conception des systèmes de commande tolérant aux défauts sont basés essentiellement sur la redondance analytique et matérielle des entrées (actionneurs) du système, [Wu 2000, Staroswiecki 2002, Khelassi 2009]. Tandis que, il est clair que la redondance matériel et analytique des sorties (capteurs) du système est très important pour l'étape de diagnostic. Peu de travaux ont considérés les deux informations d'actionneurs et de capteurs en même temps.

L'objectif de la présente thèse est de prendre en considération les informations issues de l'étape de diagnostic pour concevoir un système de FTC et d'évaluer du point de vue structurel, le niveau de tolérance aux défauts affectant le système dynamique et cela en tenant en compte les performances du système de diagnostic. Cette étude exploite un seul outil graphique qui est le bond graph. Cet outil permet de coupler les résultats du diagnostic avec les conditions de reconfiguration structurelles. De cette façon, l'ensemble de défauts critiques et non critiques du système peut être déduite.

Le bond graph est un outil de modélisation multi-physique unifié pour tous les domaines de la physique. Il est également adapté pour le diagnostic et l'analyse structurelle à cause de ses propriétés causales et structurelles. L'intérêt d'exploiter les résultats structurels repose sur le fait qu'aucune connaissance des valeurs numérique des paramètres est nécessaire à priori. Ainsi, dans le cadre de notre travail, ces propriétés seront exploitées pour la conception d'une procédure permettant l'analyse de la reconfigurabilité du système en présence de défauts. Par ailleurs, le bond graph est également utilisé pour améliorer l'isolabilité de certains défauts.

Introduction

Framework and context of the thesis

This Ph.D. thesis was prepared within the research group "Méthodes et Outils pour la Conception Intégrée de Systèmes (MOCIS)¹", of the Laboratoire d'Automatique, Génie Informatique et Signal (LAGIS - UMR CNRS 8219)². It is included in the framework of the European InTraDE project [InTraDE 2012] (Intelligent Transportation for Dynamic Environment), supported by the European commission. The project consists in designing a fault tolerant control transportation system based on the principles of intelligent autonomous vehicles, allowing handling and transporting people and goods inside confined spaces of port terminals. This work was developed under the supervision of *Mr. B. Ould-Bouamama*, Professor at Polytechnique Universitaire de Lille 1 and *Mr. R. Merzouki* Professor at Polytechnique Universitaire de Lille 1.

The research group MOCIS has an extensive experience with integrated design of systems, about modeling structural analysis, control and diagnosis through the use of a unifying tool, called bond graph (BG). Figure 1 shows the topological organization of this group. This integrated design exploits specific properties of bond graphs. These properties are (i) its causal and structural graphical aspects. (ii) Mathematical and physical properties of its behavioral model. (iii) Functional and modular topology.

In this context, several research works have been published over the last 20 years. For example, in the field of monitoring (supervision) and control analysis (framework of this thesis), the works of the group are summarized as follows: (i) Structural controllability and observability [Sueur 1991]. (ii) Monitoring properties and conditions (ability to detect and isolate faults) directly on the bond graph model [Ould-Bouamama 2003]. (iii) Development of a dedicated software for automated generation of analytical redundancy relations and fault signature matrix [Ould-Bouamama 2006].

¹<http://www.mocis-lagis.fr/>

²lagis.ec-lille.fr

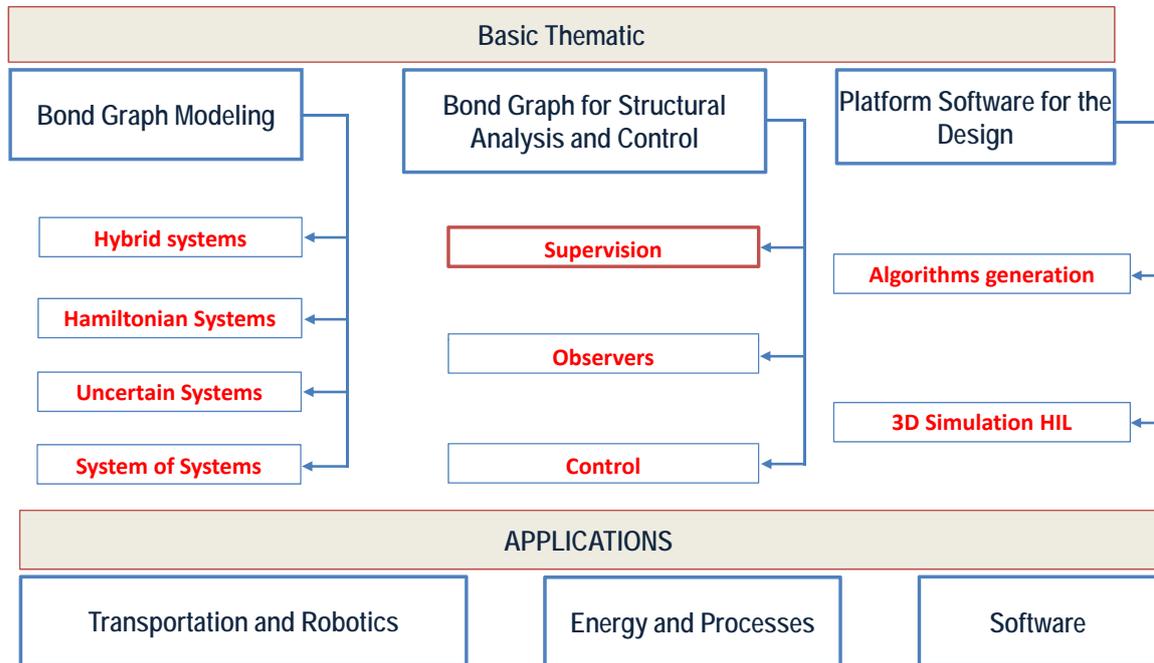


Figure 1: Organization of the works developed within the group MOCIS.

Thesis objective

The present work proposes not only a logical continuation of the referred works, but also a novel thematic within the MOCIS group. This proposition is related to the evaluation of the systems fault tolerance properties, which is a logical continuance of the monitoring studies. Thus, the main goal of this Ph.D. work is related to the **development of a structural recoverability analysis procedure based on the bond graph tool**. In this way, one can analyze the set of critical and non-critical faults prior to industrial implementation. This procedure has been tested and validated on a redundant, Heavy-Sized Intelligent Autonomous Vehicle, named RobuTAINeR.

Problematic

During the last decades the performance and quantity of control systems have been increasing substantially. Nowadays, control systems are included in several products

Introduction

that are daily used. For instance, they are usually presented in computers, cell phones, washing machines, vehicles, aircraft, etc. These control systems are designed to enable the system operating purposes to be satisfied, by assuming that the system is under normal conditions. Nevertheless, automatic systems are vulnerable to faults, which disturb their main functioning. Logically, this lack of reliability in cell phones or washing machines is not critical in the sense that the environment or humans are not in danger. However, if a fault occurs in safety-critical systems (such as, nuclear power plants and aircraft), catastrophic accidents could occur. Therefore, the concept of fault tolerant systems (FTs), raised during the 80s was initially introduced to ensure safety and accident avoidance in these safety-critical systems. This concept pointed the research world towards two new directions of investigation, which are related to fault diagnosis (FD) and fault tolerant control (FTC).

However, a current increase in complexity of modern technology systems has been leading to an increase on the frequency of faults. Obviously, it is meaningless to conceive more complex industrial systems if they are regularly jeopardizing the environment and the surrounding people. Therefore, the necessities of FTs are becoming highly required. These systems combine between FD algorithms, in terms of: (detection, isolation, and analyzing faults), and FTC strategies for control re-design. The coupling of FD and FTC is fully integrated in the design of dynamic systems in several fields of engineering, such as, automotive, manufacturing, robotic systems, etc.

The primary goal of developing FTC algorithms is to enable the system to meet with its missions requirements, while maintaining the overall safety of the system, even when subject to faults (e.g. actuators, sensors, or other components of the system), through the re-design of the control strategy. [Zhang 2008] defined FTC as:

Definition 0.2. ... FTC is a control system that possesses the ability to accommodate system component failures automatically and is capable of maintaining overall system stability and acceptable performance in the event of such failures...

With respect to the design of FTs, [Patton 1997] categorized them into two different

classes, namely Passive FTC (PFTC) and Active FTC (AFTC). The passive FTC strategies [Hsieh 2002, Liao 2002, Niemann 2005] make use of robust fixed structure controllers to make the system able to deal with a subset of faults considered at the controller design stage. On the other hand, the active FTC strategies [Mhaskar 2006, Zhang 2009, Miksch 2008] are employed after the faults are diagnosed. Thereafter, the controller adapts online to fault information, by a controller re-design mechanism. In this way, appropriate control actions are computed to stop the propagation of the fault effects. In this way, the fault effects are unable to disturb the operation of the healthy components of the system.

Different approaches have been developed for the designing and the implementation of FD procedures. These methods depend on the kind of knowledge used to describe the plant operation, and they may be broadly categorized into two groups: The approaches based on the knowledge of the system model or structure, which are referred to as *Model-based methods* [Frank 1990, Frank 1997a, Gertler 1997], and the ones relying on the availability of historical data of the process, which are referred to as *Data-based methods* [Li 2000, Srinivas 1994, Qian 2008].

The design of a FTs through AFTC strategies, is not a simple task because it should integrate both FD and FTC strategies. Moreover, not every system contains fault tolerant properties, because these properties are only founded on the principle of redundancy. Indeed, faults are only able to be diagnosed, and compensated if enough redundancy is presented in the system. Redundancy is the duplication of system components or functions with the intention of increasing reliability. For instance, measurement redundancy enables the system to observe the same physical quantity in different ways. On the other hand, actuation redundancy provides the controller with sufficient authority to proceed controlling the system in a faulty situation, probably with reduced performances [Hallouzi 2008]. In other words, redundancy enables to achieve certain functions by more than a single way, and it can be in direct or indirect forms. Direct redundancy, which is also called material redundancy, exists when identical hardware/software components are configured in parallel and available for the

Introduction

system. For example, if an actuator or a sensor fail, it would be replaced by another one with the same characteristics. An example of a system represented with a block diagram structure with direct sensor redundancy is illustrated in Figure 2, where two similar sensors measure the physical quantity y_1 . This configuration is mainly used in aircraft and nuclear platforms, where the safety requirements are very high.

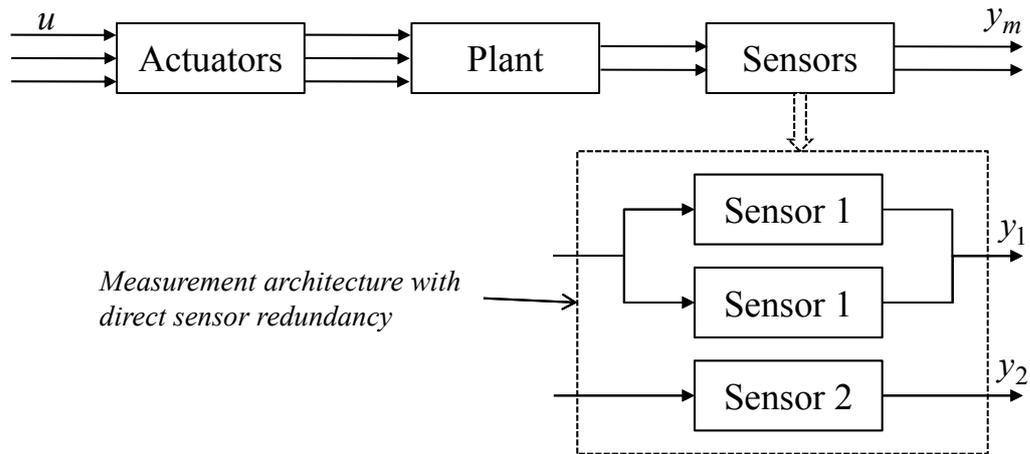


Figure 2: Block diagram representation of a system with direct sensor redundancy.

The main inconveniences of providing a system with direct redundancy is related to extra costs in terms of manufacturing, operation, and maintenance. Likewise, additional space is also required which make this approach unattractive. Consequently, exploiting indirect redundancy, also referred to as analytical redundancy, is the most appropriated configuration to conceive FTs. Analytical redundancy has to do with the ability of a system to adequate or use the available healthy components in re-planning the reconfiguration strategy with less performance and reachable objectives. This is determined by using the static/dynamic relations between variables and components of the FTs [Wu 2000]. For instance, as stated in [Kwong 1995], a system is said to have analytical actuation redundancy if in the case of an actuator break down, the control system can be automatically re-designed in such a way that adequate performance is obtained by the remaining healthy actuators. Regarding FD, redundancy is essentially exploited to compare the real behavior of the system with a reference behavior representing the system under nominal conditions.

It is then clear that a successful creation of FTs through Active fault tolerant control (AFTC) is not only dependent on the actuation redundancy, but also on the measurement one so that accurate and detailed diagnosis results are obtained. However, several contributions devoted to study the level of fault tolerance presented in the system [Wu 2000, Staroswiecki 2002, Khelassi 2009] assume that the FD step is performed, instead of also establishing their analysis on the diagnosis capabilities of the system.

Nevertheless, it is obvious that to be able to implement fault tolerant control strategies, an accurate and precise FD information is required. **Hence, the goal of this Ph.D. work is to evaluate, from a structural point of view, the existing level of fault tolerance in a system by also performing the FD procedure, which we associate with the name of *fault recoverability analysis*. This study exploits a single graphical tool (bond graph), that enables to couple the diagnosis results with fault tolerant control conditions. In this way, the set of critical and non-critical faults of the system can be deduced.** The bond graph is a unified and multi-domain tool for modeling. Due to its causal, structural, and behavioral properties, the bond graph tool is also adapted for structural diagnosis and structural control analysis [Samantaray 2008b]. The interest of exploiting the structural results relies on the fact that no knowledge of the parameter values is required. Hence, in the context of our Ph.D. work, these properties will be exploited for the design of a procedure that enables fault recoverability results to be obtained prior to industrial implementation. Moreover, the bond graph is also used to improve fault isolability. Finally, since its physical structure can be exploited for fault estimation [Touati 2012], we propose a novel way to consider the fault not as an information, but as a power. This reasoning enables to extend the results of fault recoverability and to illustrate its effectiveness, an adaptive compensation controller is designed.

Obtained results

The obtained results have made the topic of the following publications:

Journal of rank A

1. R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Bond graph model based on structural diagnosability and recoverability analysis: Application to intelligent autonomous vehicles*. IEEE Transactions on Vehicular Technology, vol. 61, no. 3, pages 986–997, 2012. [Loureiro 2012b].
2. B. Ould-Bouamama and R. Loureiro and Gautam Biswas and R. Merzouki. *Robust Graphical Methods for Diagnosis of Dynamic Systems: Review*. IFAC Annual Reviews in Control (Under revision).
3. R. Loureiro, S. Benmoussa, Y. Touati, R. Merzouki and B. Ould-Bouamama. *Integration of fault diagnosis and fault tolerant control for healthy monitoring of a class of MIMO Intelligent Autonomous vehicles*. IEEE Transactions on Vehicular Technology (Under review).

Book Chapter

1. R. Loureiro, R. Merzouki and B. Ould-Bouamama. Structural Reconfigurability Analysis for an Over-Actuated Electric Vehicle. In R. Merzouki, editeur, *Mechatronic & Innovative Applications* (www.eurekaselect.com/101969/volume/1). Bentham science, 2012. [Loureiro 2012d].

International conferences

1. R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Structural Reconfiguration Conditions Based on Bond Graph Approach: Application to an Intelligent Autonomous Vehicle*. In Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, pages 970–975, Mexico city, Mexico, August 2012. [Loureiro 2012e].
2. R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Extension of the Bond Graph Causality Inversion Method for Fault Detection and Isolation: Application to a Mechatronic System*. In Proceedings of the 8th IFAC Symposium on Fault

Detection, Supervision and Safety of Technical Processes, pages 150–155, Mexico city, Mexico, August 2012. [Loureiro 2012c].

3. R. Loureiro, S. Benmoussa, Y. Touati, R. Merzouki and B. Ould-Bouamama. *Graphical Approach for State Reconstruction and Monitoring Analysis*. The 7th IEEE Conference on Industrial Electronics and Applications 2012, pages 1205–1210, Singapore, July 2012. [Loureiro 2012a].

Outline of the thesis

This manuscript is organized in the following way:

Chapter 1: State of the Art: This first chapter is devoted to a state of the art of the existing methods in literature on FTs. The latter includes a brief introduction to FD and FTC strategies that is used to explain the problematic of fault tolerance analysis. Then, we give a detailed explanation on the works studying fault tolerance, while analyzing their qualities and drawbacks. The goal of recent research is to conclude about fault tolerance from an analytical or functional representation of the system. Nevertheless, the main problem of these techniques is that the diagnosis step is considered ideal. In our work we propose to consider the actual diagnosis information obtained from the measurement architecture of the system. In this way, we believe that a more accurate evaluation of the systems fault tolerance is obtained.

Chapter 2: Structural analysis of BG models for Fault diagnosis: This chapter details the methodology to generate Analytical redundancy relations (*ARRs*) for FDI systematically from the BG model. It proposes an extension of this method in order to generate extra non-redundant *ARRs* to isolate a higher set of faults. Finally, we exploit bond graph behavioral, structural, and causal properties to conclude if the system is diagnosable, and if using the actual diagnosis information, the fault can be structurally recovered.

Chapter 3: Local adaptive fault compensation: To exploit and extend the

structural fault recoverability results, we propose structural conditions for a local adaptive compensation that may be able to cope with a larger set of faults. The physical concept of the bond graph model enables to capture and to feed the faulty power in an appropriate location of the system. In this way, the faulty model of the system is obtained. To validate this procedure, we proposed an inverse control approach on the faulty model of the system.

Chapter 4: Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle: In this chapter the algorithms proposed throughout this manuscript are implemented and tested by the aid of co-simulations in a heavy sized intelligent autonomous vehicle (IAV), named RobuTainer, and developed in the framework of InTraDE project [InTraDE 2012].

Concluding remarks and perspectives allowing to possible further developments of this research work are provided in the end of the manuscript.

Chapter 1

State of the Art

1.1 Introduction

In our society, engineering applications are highly dependent on reliability, safety and efficiency of control systems. In the last 30 years, the field of automatic control systems has received a fair amount of attention, and therefore large improvements were achieved. However, recent studies investigate the existence of faults in these types of systems. Therefore, FD and FTC algorithms became the key to develop automatic systems that are able to continue operating, while ensuring safety, in the presence of faults on the physical components of the system. This chapter intends to establish the position of the work developed in this Ph.D. thesis in relation to previous research. To this end, we start this initial chapter by presenting an overview of the existing methods in literature, together with some definitions and concepts in the domain of fault tolerant systems (FTs).

1.2 Fault tolerant systems

As aforementioned, the strategies to achieve FTs can be divided in two categories [Patton 1997]: namely Passive Fault Tolerant Control systems (PFTCs), and Active Fault Tolerant Control systems (AFTCs). The passive approach makes use of a robust fixed structure controller to make the system able to deal with a finite and bounded set of faults that are predefined at the system design stage. This approach is easily implemented and do not demand any online information of the faults (no need of FD algorithms) and is, therefore, computationally more attractive. However, only a restrict set of faults can be considered, and they are unable to deal with unforeseen faults. In addition, in such case, the robust controller is conservative¹ thus, resulting in a low level of performance. Some of the passive FTC approaches found in literature are: Linear Quadratic (LQ) [Hsieh 2002], Linear Matrix Inequality (LMI) [Liao 2002], Reliable H_∞ controller [Niemann 2005], etc.

¹A conservative controller means that its fixed gains are computed in order to achieve system stability under a set of uncertain (faulty) parameters. Hence, losing some performance.

However, faults are effects that happen rarely. For this reason, it is not reasonable to degrade the system performance, in order to achieve robustness for a restricted class of faults [Kanev 2004]. In contrast, even if AFTC strategies are more complex for implementation from one side, they consider fault information on the other side. This fact helps to compute appropriate controllers that enable to stop the propagation of the fault effects. This is referred to as control re-design and it can be accomplished by two different strategies: fault accommodation or system reconfiguration. Hence, they are able to react to a wider range of faults and to achieve better performances for each, nominal and faulty situation [Oca 2009]. Thus, in this work, the structural analysis for the synthesis of an AFTC is produced. A general structure in the form of block diagram of an AFTC system is presented in Figure 1.1. Among the works on AFTC, one can refer to: Model Predictive Control (MPC) [Maciejowski 2003], Pseudo Inverse Method (PIM) [Miksch 2008], Eigenstructure Assignment (EA) [Zhang 2002], Reconfigurable Sliding Mode Control (RSMC) [Demirci 2005], adaptive control [Zhao 1997], etc.

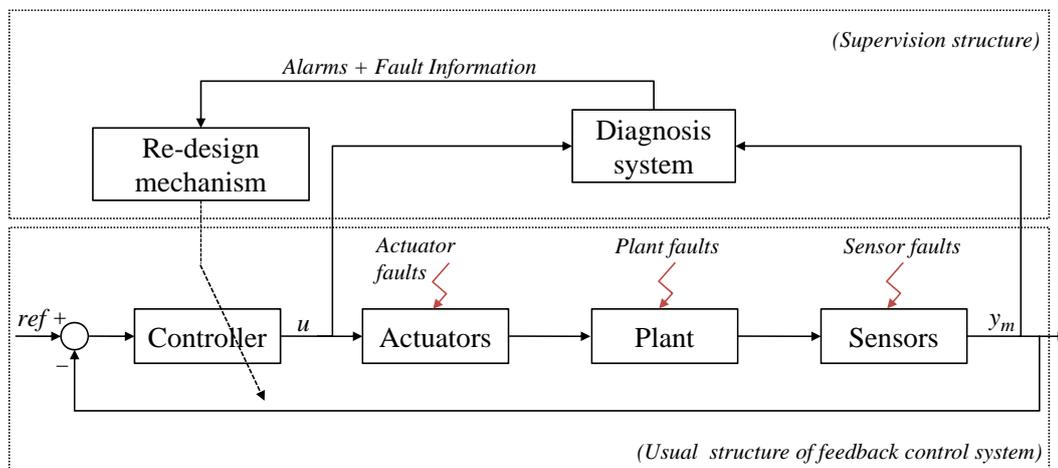


Figure 1.1: General architecture of an AFTC system.

In Figure 1.1, the solid arrows represent signals. The connection between the re-design mechanism and the controller is represented with a dashed arrow describing an information link. The AFTC structure includes a supervision structure that contains a diagnosis algorithm and a controller re-design mechanism to the usual feedback control structure. Its implementation consists of three subsystems. A controller that can be

modified according to the type and the magnitude of the fault, a diagnosis system requiring the known inputs and the measured output signals of the system, and a controller re-design mechanism which should provide the controller with the needed modifications on its parameters and/or structure.

1.2.1 Fault diagnosis

Diagnosis systems, also called monitoring systems, are essential to ensure safe operation of physical systems and to increase their reliability. A system is said to be working at his nominal conditions when it is able to furnish all the functions for which it was created with the required specifications on the system performance. Faults are undesired events that can lead to an incorrect state, and they may appear in different parts of the physical system, namely: sensors, actuators, and system plant.

Definition 1.1. (System fault). A fault in a system represents a deviation of the system structure or parameters from its nominal conditions [Blanke 2003].

- *Sensor faults:* Yields a wrong representation of the measured physical quantity. Therefore, a discrepancy between the real and the measured values is presented.
- *Actuator faults:* These types of faults can be represented by a difference between the desired input values computed by the control system and the ones furnished by the actuator output. It can be in the form of a partial or total loss of the actuator effectiveness.
- *Plant faults:* These faults include modifications in the system structure (such as a leak in a pipe,) or in the physical parameters (such as an important variation in the electrical resistance value). These types of faults induce changes in the dynamical behavior of the system.

If the deviation generated by the fault makes the component dysfunctional, its function will no longer be achieved and the fault becomes a failure.

Definition 1.2. A failure is a permanent interruption of a system ability to perform a required function.

The overall concept of the fault diagnosis consists in the following three tasks.

1. *Fault detection*: determines the occurrence of a fault in the system that leads to an undesirable behavior.
2. *Fault isolation*: intends to determine which component(s) of the system is/are not operating properly.
3. *Fault analysis*: analyses the type of the fault and its magnitude.

In real applications, the diagnosis algorithm must at least detect and isolate the fault. The analysis step is undoubtedly useful, but it may not be necessary [Chen 1999].

The task of diagnosis has inspired many studies, enabling a quick progress in this field. In a general way, all diagnosis algorithms rely on some kind of knowledge of the system. This knowledge represents a reference (model) corresponding to the normal (fault free) or abnormal behavior of the system (presence of a fault). Indeed, this reference can be obtained from historical data, or in the form of an analytical or structural model of the system. The availability of one of these models allow comparing the on-line evolution of a real process, through its measures, with its theoretical description provided by the model. The results of this comparison enable the evaluation of an abnormality in the systems behavior. Therefore, the reference model can be used as a way to validate a correct operation of the system.

Comprehensive reviews on FD can be found in the literature: [Isermann 1997, Isermann 2005, Angeli 2004, Venkatasubramanian 2003a, Venkatasubramanian 2003b, Venkatasubramanian 2003c]. Among the diagnosis methods, one can distinguish two categories. The approaches based on knowledge of the system model or structure, which are referred to as *Model-based methods*, and the ones relying on historical data of the process, which are referred to *Data-based methods*. A classification of the diagnosis methods presented in literature is proposed in Figure 1.2.

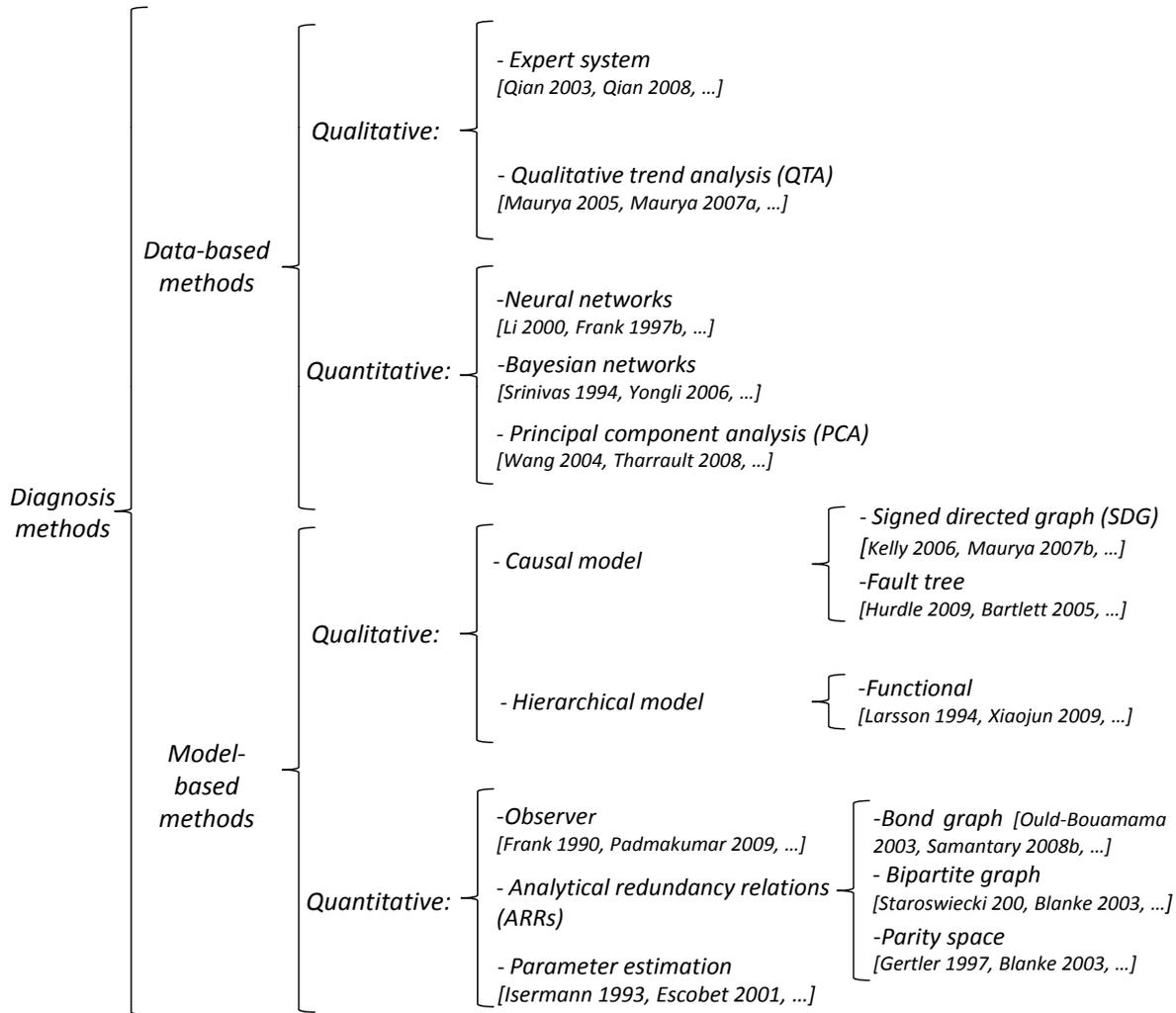


Figure 1.2: Classification of diagnosis methods.

In *Data-based methods*, the knowledge of the system is based on large amounts of recorded and collected data of the process under nominal and faulty conditions. Based on this information, different methods can be used to extract features from the process history so that it can be used as knowledge to the diagnosis system. This extraction can be done in a *qualitative* or *quantitative* way.

- Quantitative methods rely on artificial intelligence techniques, such as neural networks [Maki 1997, Li 2000, Frank 1997b], Bayesian networks [Srinivas 1994, Yongli 2006] or on statistical methods, such as principal component analysis (PCA) [Wang 2004, Tharrault 2008]. In these methods, the problem of fault

diagnosis is seen as a classification problem. In fact, it consists on solving a pattern recognition problem by classifying the data point into different predefined classes under nominal or faulty situations. Then, the objective is to determine the current class of the system, based on current observations of the process.

- Qualitative methods intend to create qualitative knowledge statements from historical data. These methods extract qualitative information from the process and are based on expert systems [Moyes 1995, Qian 2008, Qian 2003] and qualitative trend analysis (QTA) [Maurya 2007a, Maurya 2005]. Expert systems are computer programs that enable to use human expertise obtained from previous experiences to determine relations between signals and possible results as a human does. In fact, the inference procedure relies on the knowledge and experience of several operators and engineers to create a knowledge data-base, composed by a batch of (if-then) rules. The main idea of QTA is to represent measured signals as a sequence of shapes, named primitives, (such as, increasing, decreasing, constants, etc). Trend modeling can be used to explain the events occurring in a system, and diagnose faults.

These approaches have the advantage of not requiring the knowledge of an analytical or structural model of the system. They only require information collected in historical, rules, or patterns in databases. However, a lack of data compromises the ability to localize and isolate system faults, and it is difficult to obtain large amounts of historical data in different modes of faulty operation. Moreover, each diagnosis system is specific to a single process.

On the other hand, *Model-based methods* exploit the physical knowledge of the system under supervision. This knowledge is represented in the form of an analytical or graphical model of the system under nominal or faulty operation. Moreover, the system model can be created under a *qualitative* or *quantitative* forms.

- Qualitative methods directly establish relations among system variables of interest. With this methods, diagnosis can be performed from causal qualita-

tive models such as: signed directed graphs (SDG) [Chang 1999, Maurya 2007b, Kelly 2006], fault trees [Bartlett 2005, Hurdle 2009], or from hierarchical functional models [Larsson 1994, Xiaojun 2009].

Causal qualitative models are an abstraction of the system behavior, they directly establish relations among system variables of interest as causal relations, or in some cases signed causal-relations, and draw conclusions in a formal methodology. The hierarchical models exploit the means-end relations between the subsystems and describe the model at different levels. These methods are easy to develop and apply. They do not need to accurately represent the internal physical relations. Nevertheless, the quality of the method is highly dependent on the level of expertise of the developer. Finally, the diagnosis results may provide a large set of possible solutions.

- Quantitative methods base their analysis on an analytical model of the system. The idea of these approaches is to compare the real system behavior with an analytical model describing the system under nominal conditions. These comparisons yield fault indicators that are named residuals. They are computed from equations that can be solved by only using known values (inputs/outputs) of the system. Ideally, residuals are zero in normal operation and different from zero in a faulty case. Nevertheless, since a mathematical model of the dynamic system is uncertain in practice, residuals must be evaluated in order to determine the presence of a fault. This evaluation is usually done through thresholds assignments, which are only crossed in the presence of a fault. A general scheme of this procedure is presented in Figure 1.3. To compute the analytical equations, different methods can be used, such as: observers [Frank 1990, Padmakumar 2009], parameter estimation methods [Isermann 1993, Escobet 2001], and analytical redundancy relation (ARR) methods, which include the parity space [Chow 1984, Patton 1991, Gertler 1997, Hagenblad 2002], the bipartite graphs [Blanke 2003, Staroswiecki 2000], and the BGs [Ould-Bouamama 2003, Samantaray 2008b].

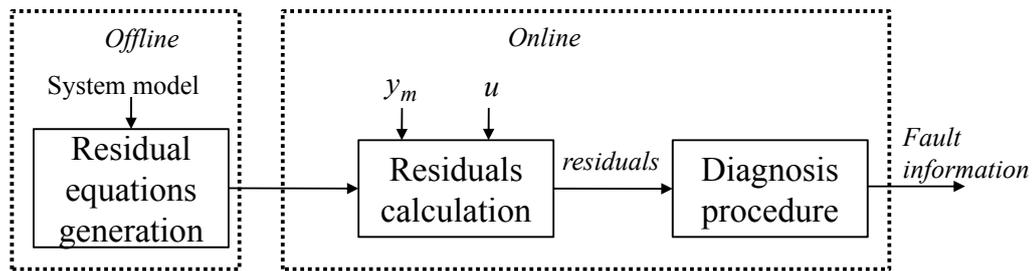


Figure 1.3: Quantitative Model-based fault detection and isolation (FDI) based on residuals.

To synthesize, these methods have some attractive properties such as: the models are based on physical principles, and they can consider transient behavior. Nevertheless, an accurate model of the system may be difficult to obtain and to model uncertainties and unknown disturbances. These may lead to false alarms. Finally, diagnosis results are highly dependent on the measurement architecture of the system.

FD is the initial step for the design of FTs. However, once a fault is diagnosed, the AFTC must react to the diagnosed fault by re-designing the control. The new control must ensure stability and acceptable performance. In the following, a brief overview on control re-design strategies is presented.

1.2.2 Active Fault tolerant control

In order to compensate the faults, a new control strategy that is able to conserve a certain level of system performance and ensuring the overall safety of the system must be used [Patton 1997]. Therefore, fault tolerance can only be achieved in the presence of a fault if both FD and FTC algorithms are successful in their functions.

Definition 1.3. (Fault tolerance). The property of fault tolerance is assigned to a system if their objectives can still be achieved, even in the case that one or more faults are affecting the system.

Regarding the control part of AFTC, fault tolerance can be achieved by two different procedures, namely fault accommodation, and system reconfiguration. The first

procedure is usually applied when the fault can be isolated, estimated, and not severe. It consists on keeping the system operating with the faulty configuration (actuators, sensors, system plant). It is performed by controller parameter adjustments, so that appropriate control actions, compensating the fault, are provided to the system. A scheme that illustrates the strategy of fault accommodation is proposed in Figure 1.4. This strategy employs exactly the same inputs and outputs of the system as in nominal conditions.

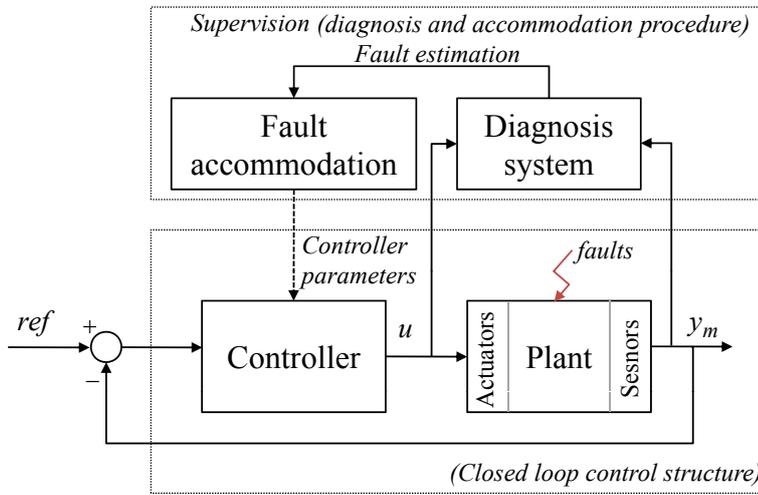


Figure 1.4: Fault accommodation scheme.

Since the structure of the controller is not changed, fault accommodation cannot counter broken control loops, i.e., it is not able to deal with severe faults that change the input/output dimensions of the plant. Therefore, system reconfiguration is usually applied when the fault cannot be estimated or if it is severe. In this case, the system is reconfigured in a way that only the healthy part of the system is used to attain the desired objectives. To exemplify this strategy, consider a Multi-Input Multi-Output (MIMO) system where the inputs $u \in \mathfrak{R}^m$ ($u(t) = [u_1(t), \dots, u_m(t)]$) and the measured outputs $y \in \mathfrak{R}^l$ is represented by the state equations (1.1).

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t). \end{cases} \quad (1.1)$$

Where, the state vector $x \in \mathfrak{R}^n$, and A , B , and C are matrices with appropriate

dimensions. If the diagnosis algorithm detects a failure at instant t_f in the first actuator that it becomes uncontrollable, then the new input vector $u \in \mathfrak{R}^{m-1}$ will be expressed as: $u_f(t) = [u_2(t), \dots, u_n(t)]$, and equation (1.1) becomes (1.2) at $t \geq t_f$.

$$\begin{cases} \dot{x}(t) = Ax(t) + B_f u_f(t), \\ y(t) = Cx(t). \end{cases} \quad (1.2)$$

In this case, the system structure is the same as in the faulty free situation, but there is a reduction on the dimension of the input vector. This is the logic of the system reconfiguration strategy, where the set of inputs/outputs are modified in order to control the faulty system. A scheme that illustrates this strategy is proposed in Figure 1.5.

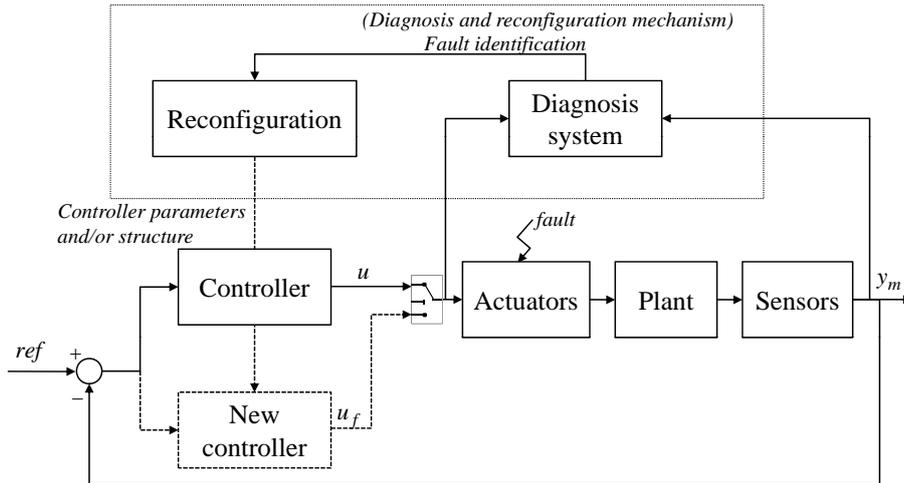


Figure 1.5: System reconfiguration scheme.

An interesting bibliographical review of active fault tolerant control systems can be found in [Zhang 2008]. The paper proposes a classification of the different methods based on the design approaches and on their applications. In addition, open topics of research are also discussed. For clarification purposes, three methods of AFTC are overviewed.

Pseudo Inverse Method (PIM):

This technique uses the concept of *model-matching*, i.e., it intends to keep as similar as possible the reconfigurable and the nominal closed-loop systems. PIM has been often

studied for FTC problems [Gao 1991, Miksch 2008]. The main idea is presented in the work [Gao 1991] in the following way: consider the state space model of the nominal system represented as in (1.1). The closed-loop system is obtained by the state feedback control law $u = Lx$. Then, assuming that the system is working under the effect of a fault, the new state space model can be represented as:

$$\begin{cases} \dot{x}_f = A_f x_f + B_f u_f, \\ y_f = C_f x_f, \end{cases} \quad (1.3)$$

and a new state feedback control law $u = L_f x_f$ is required. Hence, the objective is to update the feedback gain L_f , in such a way that, the closed-loop of the reconfigured system approximates the nominal one. In [Gao 1991] this gain matrix is obtained by minimizing the Frobenius norm $\|\cdot\|_F$ of the closed-loop systems differences:

$$J = \|(A + BL) - (A_f + B_f L_f)\|_F \quad (1.4)$$

This method has the advantage of its simplicity and of having a low online computational effort [Miksch 2008]. Its drawbacks are related with the fact that the stability of the reconfigured closed-loop system is not always guaranteed [Gao 1991], and actuator limits are not taken into consideration to compensate the fault effects [Miksch 2008]. To overcome the problem of the stability of the reconfigured closed-loop system, the contribution of [Gao 1991] took in consideration the stability constraints while updating the feedback gain. This approach is named the *modified pseudo-inverse method*, and it becomes a constrained optimization problem. Hence, the computational complexity increases.

Eigenstructure Assignment (EA):

EA is one of the most used techniques to control MIMO systems [Zhang 2002]. Among the works in which this technique was implemented for FTC systems, one can refer to [Zhang 2001, Zhang 2002]. It is known from control theory that the eigenvalues and eigenvectors describe the stability and dynamic behavior of a closed-loop system [Zhang 2001]. Thus, the aim of this approach is to obtain the eigenstructure (eigen-

values and eigenvectors) of the reconfigured closed-loop system as similar as possible to the nominal one. This is obtained by reconfiguring the feedback control gain when a fault is detected in a system. In [Zhang 2002], a detailed explanation regarding the algorithm used to update the feedback control gain, is given. As a disadvantage, it can be stated that the system performance is not optimal [Zhang 2001].

Model Predictive Control (MPC):

MPC is an effective technique for solving problems of multivariable constrained optimal control. In MPC, an internal model of the plant is used to predict the system behavior of a future finite time horizon. Based on these predictions, a cost function is minimized over a sequence of the future input commands. The first of such sequence is applied to the plant and, at the next time step, the optimization is repeated over a shifted horizon. The works of [Maciejowski 2003, Mhaskar 2006, Theilliol 2009, Miksch 2008] are based on the principle of MPC in the area of FTC.

This controller can be easily adapted to faults. For instance, actuator faults can be represented by modifying the constraints of the optimization problem, and sensor faults can modeled by modifying the internal plant of the system [Maciejowski 2003]. In addition, due to the possibility of defining constraints, this method ensures that actuators saturation does not occur. The basic idea of the MPC formulation is presented under.

Consider the following cost function:

$$J(k) = \sum_{i=1}^{H_p} \|y_{tr}(k+i|k) - y_{ref}(k+i|k)\|_Q^2 + \sum_{i=0}^{H_u-1} \|\Delta u(k+i|k)\|_R^2, \quad (1.5)$$

hence, the objective is to compute a control input which minimizes (1.5), subject

to the dynamic model of the system, and to the following constraints:

$$y_{tr(min)} \leq y_{tr}(k+i|k) \leq y_{tr(max)}, \quad i = 0, \dots, H_p - 1 \quad (1.6a)$$

$$u_{min} \leq u(k+i|k) \leq u_{max}, \quad i = 0, \dots, H_u - 1 \quad (1.6b)$$

$$\Delta u_{min} \leq \Delta u(k+i|k) \leq \Delta u_{max}, \quad i = 0, \dots, H_u - 1 \quad (1.6c)$$

$$\Delta u(k+i|k) = 0, \quad i = H_u, \dots, H_p - 1 \quad (1.6d)$$

where, $y_{tr}(k)$ is the vector of estimated variables to be controlled, $y_{ref}(k)$ is the desired reference, $u(k)$ is the controlled input, and $\Delta u(k) = u(k) - u(k-1)$. Q , and S are suitable weighting matrices. Finally, H_p and H_u are the prediction and controlled horizons, respectively.

In [Maciejowski 2003] the validation of this approach is done by proving that a plane crash is avoided by using the MPC-based FTC. In [Mhaskar 2006] the same approach for nonlinear systems with model uncertainties is presented, and it is applied to a chemical reactor example. The problem with MPC arises from the fact that an optimization process is required at each sampling instant thus, often making the problem computationally complex, mainly for nonlinear systems. The stability proof is a very difficult issue in MPC and research on it remains being performed. In addition, online tuning strategies also remain a complicated topic and may run into infeasibility. Especially in the case of modeling uncertainties, hard constraints and large disturbances [Miksch 2008].

In this section we presented some of the existing methods in the literature for the development of FD algorithm and FTC control strategies. Large quantities of work have been directed to these issues. However, only few attempts are known for focusing in the evaluation of systems fault tolerance. Indeed, the design of FD and FTC strategies is highly related with the redundancy presented in the system in terms of both measurement and actuation. Hence, in the following section, we will propose a review on the works based on the analysis of these fundamental properties.

1.3 Control reconfigurability analysis and problem statement

The previous section explained the two different strategies for AFTC. However, fault accommodation or system reconfiguration can only be applied if sufficient redundancy is presented in the system. Indeed, fault accommodation requires the estimation of the faulty parameter thus, measurement redundancy is required. For system reconfiguration an alternative set of I/O is necessary to control the faulty system. Hence, both measurement and actuation redundancy is need. It is clear that, a system without adequate redundancy cannot be made to effectively tolerate faults regardless of the control strategy employed [Shaker 2011]. Therefore, for analysis and synthesis of FTC system design, some research works given in the literature proposed fault tolerant measures (i.e., the quality and capacity of the system to cope with faults) that can help the elaboration of supervisory strategies including the FD algorithms. [Wu 2000] initiate the studies of these fault tolerant measures and referred to it as *control reconfigurability*. Consequently, its idea was further developed and different results were proposed. We briefly review some of the methods reported in the literature.

1.3.1 Control reconfigurability analysis

As aforementioned, a system can tolerate faults if and only if redundancy is available. Therefore, these works rely mainly on the study of this redundancy. In [Wu 2000], a Gramian based reconfigurability measure, is proposed for linear time-invariant systems. In this work, the measures of fault tolerance for a given system rely on the notion of smallest second-order mode (σ_{min}) of its model. This measure is based on the idea that the quantity of redundancy in the model with respect to a particular fault scenario can be assessed by the decrease in the value of smallest second-order mode from the nominal situation. This concept is established based on the work regarding controllability and observability performed in [Moore 1981]. In this work, it is stated that under a specific state transformation the observability and controllability Gramian

are equal $W_c = W_o = \text{diag}\{\sigma_1, \dots, \sigma_n\}$, $\sigma_i \geq \sigma_{i+1}$. Furthermore, a minimal second order mode value is defined as the limit in which the system respects fault tolerance requirements. Consider a linear system in the form:

$$\begin{cases} \dot{x}(t) = A(\theta)x(t) + B(\theta)u(t), \\ y(t) = C(\theta)x(t), \end{cases} \quad (1.7)$$

where, $\theta = 0$ under normal conditions and varies in the presence of faults. Moreover, consider S as a subset in Ω , and let Ω be the set over which θ belongs under faulty conditions. Hence, the level of fault tolerance (ρ_s) over S is given by:

$$\rho_s \equiv \min_{\theta \in S \subseteq \Omega} \sigma_{\min}(\theta) \quad (1.8)$$

Finally, the second order modes are calculated for each input fault. If its minimum value is smaller than a pre-defined threshold (ρ_{\min}), control reconfigurability is not feasible. Therefore, the system is fault tolerant if the following holds:

$$R \equiv \{\theta \in \Omega | \sigma_{\min}(\theta) \geq \rho_{\min}\}. \quad (1.9)$$

The main problem of this work is that it verifies the performance of the system actuation and measurement scheme itself. Hence, instead of evaluating the systems fault tolerance, it provides a measure of its admissibility with respect to energetic constraints.

Thereafter, [Staroswiecki 2002] proposed to tackle this issue in the context of the control problem. In this work, two main constraints are proposed: 1) The control energy required is of no importance giving that the system objectives are achieved in spite of the fault, 2) The energy required is too high, and does not pay up for a fault tolerant strategy, even if the system remains controllable. In this work, the system fault tolerance is concluded by evaluation of the system's energy consumption. To this end, consider the system given in (1.1). The criterion used to minimize the energy

consumed by the system is the following (1.10):

$$J(\gamma) = \int_0^{\infty} \| u(t)^2 \| dt. \quad (1.10)$$

Where γ is the systems objective. This equation represents the minimum energy required to take the system state from $x(0) = \gamma$ to $x(\infty) = 0$. The solution of (1.10) is known from control theory and it can be written as follows:

$$J(\gamma) = \gamma^T W_c^{-1} \gamma. \quad (1.11)$$

Therefore, it can be concluded that the actuators performance is based on the control objectives (γ) of the system. To overcome this problem, the control objectives can be disregarded when the worst case scenario in terms of energy consumption is evaluated.

Definition 1.4. In the presence of faults, the actuators performances are characterized independently to the control objective by the maximum eigenvalue of $W_c^{-1}(I)$, which is interpreted as the maximum energy required to transfer the system state to its origin. This value of energy corresponds to the worst case, that can occur under degraded functional conditions [Blanke 2003].

$$Q(I) = \lambda_{max}[W_c^{-1}(I)], \quad (1.12)$$

where Q is defined as the maximum energy which might be required to transfer the system state from $x(0) = \gamma$ to $x(\infty) = 0$, for some $\gamma \in \mathfrak{R}^n$. I is the set of actuators, W_c is the controllability Gramian, and λ is the eigenvalues of matrix $W_c^{-1}(I)$.

Then, to evaluate the fault tolerance property, this problem is studied for different actuator subsets $I_s \subset I$ in order to obtain the set of recoverable failures. Finally, if the maximal necessary energy of a given subset ($\sigma(I_s)$) is bigger than a predefined threshold, it is concluded that fault tolerance cannot be achieved under desired energy specifications. Finally, even if in [Staroswiecki 2002], the problem of fault tolerance results is performed from the control problem. The obtained results are essentially the

same as the ones in [Wu 2000]. Indeed, both results rely on the concept of controllability Gramian.

Some extension of the work in [Staroswiecki 2002] were proposed in [Khelassi 2009]. This work proposed a reconfigurability index and actuator reliability analysis. The reconfigurability index (ρ) is based on the maximum energy consumed after fault occurrence as described in (1.13).

$$\rho(I_s) = \left(\frac{\lambda(I_s) - \lambda_{min}}{\lambda_{max} - \lambda_{min}} \right). \quad (1.13)$$

Where λ_{max} is the upper value of energy in the worst degraded functional case. λ_{min} is the maximum value of energy required in the nominal case, while $\lambda(I_s)$ is the value of energy under considered faulty conditions. The index (1.13) can be seen as a representation of the control performance degradation according to the value of energy consumption under degraded modes.

Definition 1.5. The system can still be controlled under acceptable performance in faulty case if:

$$\rho \leq \rho_{thr}, \quad (1.14)$$

where ρ_{thr} is a predefined threshold representing the maximal acceptable degradation of a control solution.

In addition to this reconfigurability index, [Khelassi 2009] also proposed reliability analysis of system inputs to ensure that the new configuration can achieve the system goal until the mission ends. The idea is to base the computation of the acceptable energy threshold (ρ_{thr}) on reliability analysis. This is based on actuator failure probabilities, which are updated over time in function of the intensity of control in the actuators. Reliability is defined as the probability that units, components, equipment and process will achieve their goal for a specific period of time in its expected environment.

The presented works are applied to linear systems, therefore [Yang 2006] performed reconfigurability analysis, based on the controllability concept, for a class of linear hybrid systems. This system exhibits both continuous and discrete dynamic behavior. In this work, faults are classified in three classes: qualitative (one mode of the system is lost), quantitative (fault that only affects one mode of the system without losing it), and hybrid (faults that cause a loss of one mode of the system and also affect the dynamics of the remaining modes). Then, the rank of the controllability matrix for hybrid systems is verified for different sets of faults. One issue of this work is related to the fact that the performance indexes from the previous works are not considered.

Bilinear system¹ analysis for reconfigurability were given in [Shaker 2011]. In this work, they use exactly the same logic of the second order modes as in [Wu 2000]. Moreover, [Aitouche 2005] proposes to study nonlinear controllability, in terms of the Lie algebra [Isidori 1995], of a system under actuator failures. The basic idea is to identify the minimal and redundant sets of actuators keeping reachability unchanged.

The previous techniques are all implemented offline hence, [Contreras 2011] proposed a method that evaluates online control reconfigurability through input/output data.

Furthermore, reconfigurability was also studied from functional representations of the system, rather than analytical ones. Among this works, one can refer to, generic component models (GCM) [Gehin 2008, Staroswiecki 1998], and multilevel flow models (MFM) [de la Mata 2010], and discrete event systems [Dangoumau 2005].

MFM is a modeling tool, introduced in [Lind 1990], used to model dependencies in a process, i.e., how the variables in a process are influenced by each other. MFM employs a hierarchical graphical scheme to represent the relations between goals, functions, and physical components of a system. The goals denote the objectives of the system. The functions are the system capabilities that may (or may not) be performed by the

¹Bilinear systems are a particular kind of nonlinear systems. They are linear in the state and linear in control, but not jointly linear in both

physical components. Functions are connected in terms of flow of mass, energy, or information [Ohman 1999]. The basic idea is to recognize all goals of the system, and functions that are required to perform each goal. On the other hand, GCM describes a system based on the services provided by its components. The services are organized into subsets with respect to a given situation and set of objectives to be achieved.

These models are a functional representation of the system and they describe the system from the user's point of view. These functional descriptions enable reconfigurability analysis to be performed by verifying the different ways in which goals/objectives can be achieved. Nevertheless, functional representations require a highly detailed functional description of the system, which increases exponentially with the size of the process. Moreover, there is also no algorithm to validate the functional models, thus their accuracy is highly dependent of the engineers/developers knowledge of the process.

A comparative table of the aforementioned techniques studying the level of fault tolerance is presented in Table 1.1. This table synthesizes the different works in literature that study the analysis of reconfigurability. Hence, we try to provide a set of evaluation factors that can position our work with respect to the referred ones. This table is given in terms of offline and online analysis. FD analysis, meaning the capabilities of the measurement architecture for FD is considered. Type of faults, of the model, and the controller re-design strategy considered. Moreover, energy considerations indicate whether the approach considers or not the energy required to control or to observe the system. Finally, the methods are also distinguished between the different types of analysis, such as graphical and analytical.

Remark. The work [Loureiro 2012b], can be applied to nonlinear systems if thee nonlinearities are modeled in a bond graph model as known modulated sources.

Table 1.1: Comparison of different procedures for control reconfigurability analysis.

Reference	Off-line	On-line	FD analysis	Type of faults considered	Model used	Control re-design strategy	Energy consid.	Type of analysis
[Wu 2000]	✓	×	×	S, A	L	SR	✓	
[Staroswiecki 2002]	✓	×	×	A	L	SR	✓	
[Aitouche 2005]	✓	×	×	A	NL	SR	×	An
[Yang 2006]	✓	×	×	A	H	SR	×	
[Khelassi 2009]	✓	×	×	A	L	SR	✓	
[Contreras 2011]	✓	✓	×	A, S	L	SR	✓	
[Shaker 2011]	✓	×	×	A	BL	SR	✓	
[Gehin 2008]	✓	×	×	S, A, P	L, NL	SR	×	
[Dangoumau 2005]	✓	×	×	A, S	L, NL	SR	×	
[de la Mata 2010]	✓	×	×	S, A, P	L, NL	SR, FA	×	
[Loureiro 2012b]	✓	×	✓	S, A, P	L, NL	SR, FA	×	

✓ means available, × not available. *Type of considered faults:* Sensor faults-noted **S**, Actuator faults

-noted **A**, Plant faults-noted **P**. *Type of model:* Linear (**L**), Hybrid (**Hy**), Bilinear (**BL**), Nonlinear

(**NL**), Graphic (**Gr**). *Controller re-design strategy:* fault accommodation (**FA**), and system

reconfiguration (**SR**). *Type of analysis:* Analytical - noted **An**, graphical - noted **Gr**

1.3.2 Problem statement

As presented in Table 1.1, researches devoted to the reconfigurability analysis of systems can be divided in two categories, based on the knowledge used for formulating fault tolerance measures. The ones applying analytical analysis from an analytical representation of the system [Wu 2000, Staroswiecki 2002, Yang 2006, Khelassi 2009, Contreras 2011, Shaker 2011], and the others exploiting the graphical properties of the continuous and of the discrete event system model [Dangoumau 2005, Gehin 2008, de la Mata 2010]. Most of the analytical works tackle the problem of fault tolerance with respect to energetic and reliability constraints. On the other hand, the approaches exploiting functional information obtained from the graphical models intend to verify the different ways to achieve the same goal/service.

We remark that all the presented works investigate fault tolerance regardless of the diagnosis information, i.e., their results are based on scenarios in which fault diagnosis is considered ideal. However, even if the objective of these works is to analyze the level of the system fault tolerance, the diagnosis performances must be considered. Indeed, even if a system is highly redundant, FTC strategies are unable to be implemented if reliable fault information cannot be obtained. Moreover, since diagnosis analyses are not performed, the controller re-design strategy is only performed through system reconfiguration. In fact, these methods cannot furnish any indications regarding the selection of the control re-design approach (fault accommodation or system reconfiguration). Most of the works are particularly appropriated for offline analysis, and only consider sensors (y_f) and actuator (u_f) faults. A general scheme that represents the idea of most methods previously presented in the form of block diagram is proposed in Figure 1.6. We notice that in this scheme, the diagnosis is assumed ideal (box in grey) and only sensor and actuator faults are considered.

Hence, the aim of our work is to tackle the problem of fault tolerance from another perspective. We do not rely simply on the ability to control the system regardless of diagnosis capabilities. Actually, both actuation and measurement redundancy are inherent properties of the system. Therefore, we also base our fault tolerance evaluation

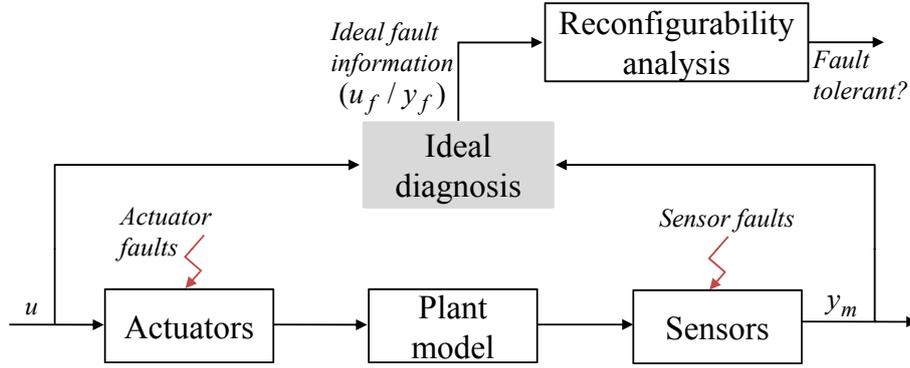


Figure 1.6: General scheme of fault recoverability analysis found in literature.

on the diagnosis results. In this way we believe that a more accurate and reliable fault tolerance analysis can be achieved when studying process applications. Since we also consider the fault information, in this thesis we define the measures of fault tolerance as *fault recoverability*. The fact that we perform FD, enables to consider both fault accommodation and system reconfiguration. In addition, our analyses do not rely simply on actuator and sensor faults, but also on the plant ones. A general block diagram on Figure 1.7 describes the main contribution of this work based on the study of fault recoverability. In this scheme, the diagnosis information obtained from the measurement architecture of the system is considered (box in grey). Moreover, the faults can be located in actuators, sensors and in the system plant.

To accomplish this challenge, the bond graph tool is going to be applied.

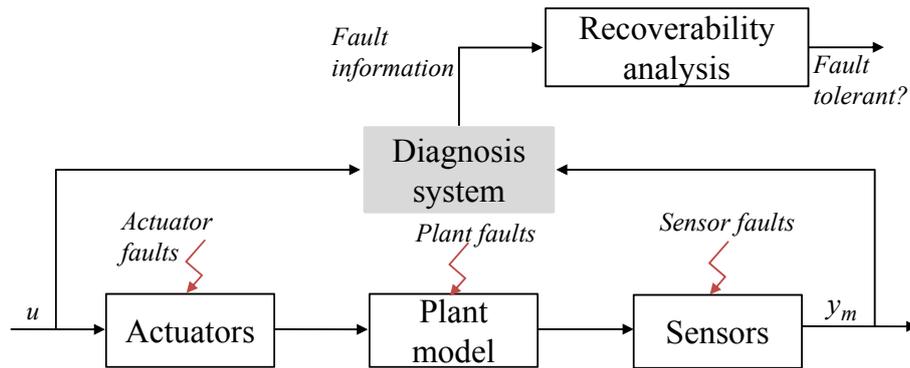


Figure 1.7: Proposed scheme of fault recoverability analysis. In this case, the real diagnosis information is considered, together with plant, actuator and sensor faults.

Definition 1.6. Fault recoverability is the ability of the system to attain its objectives

in the presence of faults, given that the fault diagnosis is able to furnish appropriate results to the controller, and enough redundancy is presented in the system.

1.4 Why use the bond graph tool for fault recoverability analysis?

The design of a valid model of the system is often the first step for designing engineering systems. A model that is able to analyze and reproduce the behavior of the system is highly beneficial when testing different types of algorithms, such as controller, observers, diagnosis, etc. Hence, the human expert usually chooses the modeling methodology better adapted to the current problem. The bond graph formalism, which was invented by *Paynter* [Paynter 1961], is a topological modeling language, where the energy exchanging between the components of a physical system are captured in a graphical form. This exchange represents the fundamental notions of physics, and the modeling reasoning consists on associating to each physical component of the system a basic element representing its physical phenomena. This methodology is domain-general, and it has been applied to a number of modeling applications that include electromechanical [Van-Dijk 1994], thermofluidic and thermal [Thoma 2000], and chemical and thermochemical [Karnopp 1990] systems. Actually, four levels of modeling can be represented by a bond graph model [Samantaray 2008b]. The *technological level* consists on the creation of the so called word bond graph. This uses words to represent large models in a hierarchical manner and it provides a simple and comprehensive way to describe the main components of the system and their interactions in terms of power and information. The *physical level* treats each component as an object that is modeled as a sub-graph structure composed of basic bond graph elements and a set of ports that are used to connect with other objects. Here the BG is used as a unified language for all domains of physics. There is also the *mathematical level* and as the name says, it is concerned with the creation of a mathematical model of the system composed of algebraic and differential equations. This is obtained by writing

the constitutive equations of the components and constraints. Finally, the *algorithmic level* exploits the concept of causality to obtain mathematical model in a systematic fashion from the BG model of the system.

Indeed, by making use of the BG tool, the obtained model of the system contains several attractive properties such as: behavioral, structural, and causal that can be exploited not only for modeling, but also for analysis and synthesis. These models can also be an excellent support to study systems supervision [Samantaray 2008b, Ould-Bouamama 2003]. The graphical and causal properties of the BG enable to analyze the structural results of FDI before industrial design, and in a second phase it is able to generate algorithms for online diagnosis. This is why we use the BG as a unified tool for system recoverability analysis. Indeed, from the same tool we are able to obtain structural FDI that the considered to analyze structural recoverability. In this way, to narrow the bridge between FDI, and fault tolerance measures. Further explanations on the BG tool for dynamic modeling are given in Appendix A.

1.5 Conclusions

In this chapter we detailed the concept of FTs. To achieve this goal, a short explanation on the logic behind the different FD and FTC approaches was provided. Moreover, the concept of fault tolerant measures was also detailed. Most part of existing methods in literature uses an analytical representation of the system. These works use analytical models under state space format and assume that the FD step is performed. Moreover, its main idea is to verify systems input or output redundancy, and if the required energy to perform reconfiguration strategies is acceptable. We have noticed the existence of an incoherence when examining the level of fault tolerance of a system that is related to the fact that it is only performed from a control perspective, either in terms of actuation or of measurement. However, since AFTC required both FD and FTC, we believe that more appropriate results are obtained if the diagnosis results are also considered to assess the level of systems fault tolerance. **Furthermore, since**

analytical approaches require accurate models, and numerical values of the parameters, which are not always available in real systems, we propose to use the bond graph. This will be used not only for modeling, but also for diagnosis, for system recoverability analysis, and for synthesis of fault compensation.

Chapter 2

Structural recoverability analysis from a bond graph model

2.1 Introduction

The first step to evaluate the level of fault tolerance of a physical system is related with the design of its dynamic model. There are different modeling methods such as: analytical (state space) and graphical (bond graph, bipartite, digraphs, signed digraphs). Among them, BG models serve as knowledge of a large amount of structural, functional, and behavioral information and their relationships. Hence, this representation enables conclusions to be made about the system from a structural point of view, i.e., without knowing their numerical values. Its causal structure was initially exploited to determine structural conditions of controllability and observability [Sueur 1991], diagnosability [Ould-Bouamama 2003], and also for static and dynamic decoupling [Feki 2008]. However, far too little attention has been paid to control re-design analysis e.g., [Samantaray 2008a]. The aim of this chapter is to propose a methodology to perform structural analysis of fault recoverability [Loureiro 2012b]. The idea is to exploit the behavioral, structural, and causal properties of the BG tool to conclude, from a structural point of view if the system is diagnosable. Once this information is obtained, it is conclude for which set of faults the system is structurally recovered.

2.2 Interest of structural analysis

In the context of modeling, control synthesis and fault diagnosis, most results are usually dependent on the systems parameters. This fact prevents from obtaining valid information about the system at an early design stage. In addition, once a parameter is modified, a new analysis phase must be conducted in order to verify if the results on systems performance remain valid. This is where the role of structural analysis is introduced. Indeed, structural analysis enables results to be obtained by analyzing the structure of the system information, and therefore it defines necessary conditions that are valid for most values of numerical parameters. A structural description of a system is based on the existence or not of a link between variables and constraints of the system, which is refereed to as structural graph [Blanke 2003]. This analysis relies

on the fact that some system relations do not change such as:

- Relations expressing connections between subsystems,
- Energy conservation laws,
- System inputs and outputs,
- ...

Structural analysis is usually performed during the systems design phase and they enable to deduce a variety of structural properties, such as: system controllability, observability, diagnosability, etc. Once it is performed, the designer receives a simple set of exploitable information that is obtained from the system structure. A definition on structural analysis in the context of BG model is given as follows:

Definition 2.1. A property of a system is said to be structural if: [Rahmani 1993]

- It only depends on the types of elements (bond graph) composing the system, and on the way that they connect between each others regardless of their numerical value.
- It is verified for most values of the parameters.

Hence, exploiting the structural analysis enables to perform initial assessments on systems without knowing their numerical values of the physical parameters. The results are valid for most of the numerical values. In addition, structural analysis provides a better understanding of the systems behavior and furnishes exploitable results previous to industrial implementation. A structural representation of the system is mainly presented in the form of a graph. Among the graphical representations in which structural analysis can be performed, one can refer to the bipartite graphs [Blanke 2003], bond graphs [Samantaray 2008b], and linear graphs [Dion 2003, Boukhobza 2007].

2.3 The fault tolerant control problem

In order to detail the structural recoverability analysis procedure, the problem of fault tolerant control is introduced in this section. As suggested in [Staroswiecki 2008], the

works on FTs found in the literature do not rely on a unified vocabulary. Therefore, the definitions used in the works [Blanke 2003, Staroswiecki 2008, Staroswiecki 2001] are considered in this Chapter. In the referred works, a control problem is defined by $\langle \Sigma_o, C(\theta), U \rangle$, where Σ_o and U are respectively, the system objectives and the set of admissible control laws. Moreover $C(\theta)$ are the constraints C that represent the system behavior, while θ is the parameter that C depends on. When a fault occurs in a system, the control problem changes, and as a consequence, these changes must be studied to keep achieving the Σ_o . The FDI and the fault estimation algorithms perform these studies.

Definition 2.2. (Objectives) : System objectives (Σ_o) are a set of specifications, which the system Σ should respect, where $\Sigma_o = \{o_1, o_2, \dots, o_k\}$, where k is equal to the total number of Σ_o .

Definition 2.3. (Recoverable fault) : A fault is recoverable if Σ_o can still be achieved after solving the faulty control problem $\langle \Sigma_o, C_f(\theta_f), U_f \rangle$ [Staroswiecki 2008].

Definition 2.4. (Fault accommodation) : This strategy intends to hold the faulty system (Σ^f) operating when the Fault Estimation (FE) algorithm furnishes an estimate of the fault so that the estimated model of the faulty system ($\Sigma^{\hat{f}} = (\hat{C}_f(\hat{\theta}_f), \hat{U}_f)$) is defined. In this case, the controller parameters are adapted to $\Sigma^{\hat{f}}$ (solves the control problem $\langle \Sigma_o, \hat{C}_f(\hat{\theta}_f), \hat{U}_f \rangle$).

Definition 2.5. (System reconfiguration) : System reconfiguration is a strategy in which Σ^f is modified by disregarding faulty components so that only the healthy part of the system (Σ') is controlled in order to achieve the desired Σ_o (solves the "new" control problem $\langle \Sigma_o, C'_n(\theta'_n), U'_n \rangle$). The unknown faulty dynamics can not affect Σ' .

The procedure for structural recoverability analysis mainly relies on the following steps:

Step 1: Structural information regarding fault detection, isolation and estimation is obtained from the BG model and Fault Signature Matrix (FSM).

Step 2: Based on the location of the fault, different structural analysis can be performed. It is possible to consider sensor, actuator, plant, and non-isolable faults.

Step 3: Verify structural properties of controllability, observability, and monitorability. In the case of non-isolable or plant faults, verify if the unknown dynamics of the faults can be removed from affecting the system objectives.

2.4 Bond graph model for structural fault diagnosis analysis

From a structural point of view, a dynamical model of any physical system can be represented by the pair (C, Z) , where $C = \{c_1, c_2, \dots, c_N\}$ is a set of constraints, and $Z = \{z_1, z_2, \dots, z_M\}$ the set of system variables, which is composed of two subsets: the unknown $\{X\}$ and known variables $\{K\}$, $Z = \{X\} \cup \{K\}$. The bond graph representation as a tool for modeling, analysis, and synthesis has proven to be interesting to perform structural analysis.

2.4.1 Bond graph representation

The bond graph formalism is a graphical modeling language that makes the modeling systematic by following the propagation of the flow or the effort between interconnected components of physical systems. It is a unified modeling tool based on the physical behavior of the system, considering that any dynamic is defined as a set of basic elements that exchange power between them. It generalizes modeling of physical components for multi domains (such as, mechanical, electrical, thermal, etc.). In addition, it captures the functional, behavioral and structural aspects of the physical system.

Definition 2.6. A bond graph, denoted $G(S, A)$ is a unified graphical language for multi-physic domains. It is composed of a set of vertices S , representing physical components, subsystems, and other basic elements called junctions. While the edges

Structural recoverability analysis from a bond graph model

A , called power bonds represent the power exchanged between nodes. This power is labelled by two conjugated power variables, named effort (e), and flow (f).

In Figure 2.1, the concise notion of BG is mentioned. This notion gives four information: The existence of a physical link between subsystems or components (bond), the type of power, which is obtained by the power variables, the power direction from the half arrow and the causality. In BG notation, the physical components are modeled

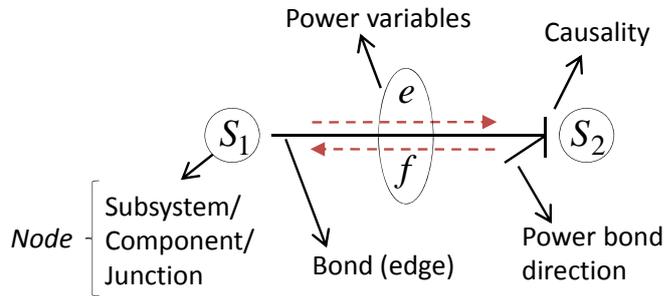


Figure 2.1: General representation of a bond graph model.

from the following basic elements: The passive elements R that dissipate energy, while the C , and I store energy. Se , Sf , MSe , and MSf are active elements that provide power to the system, while power conserving elements are $(0, 1, TF, GY)$.

The set of constraints C on a BG model are assigned from the behavior (C_B), the structure (C_S), the measurements (C_M), and the control system (C_U). Thus, C is defined as follows:

$$C = \{C_B\} \cup \{C_S\} \cup \{C_M\} \cup \{C_U\} \cup \{C_A\}. \quad (2.1)$$

The behavioral equations (C_B) are obtained from physical laws, i.e., the constitutive equations of the BG elements, associated to the behavior of the elements R , I , and C , namely C_R , C_I , and C_C (2.2).

$$C_B = \{C_R\} \cup \{C_I\} \cup \{C_C\}. \quad (2.2)$$

The structural constraints (C_S) capture the conservation laws (mass, energy, etc.).

They are deduced from the junction equations (0 or 1), noted C_0 and C_1 , and from transformers TF , and gyrators GY presented in the BG model, namely C_{GY} and C_{TF} (equations 2.3).

$$C_S = \{C_0\} \cup \{C_1\} \cup \{C_{GY}\} \cup \{C_{TF}\}. \quad (2.3)$$

The measurement constraints (C_M) capture the relations between state variables and the output signals, which are associated with sensors. In BG models, the sensors are represented as detectors of flow (Df), and detectors of effort (De) (equations 2.4). The detectors Df are placed on 1-junctions, while the detectors De are placed on 0-junctions.

$$C_M = \{C_{m_f}\} \cup \{C_{m_e}\}. \quad (2.4)$$

The constraint (C_U) represents the control algorithm. The control laws consider the measure signals Y_m , the desired outputs Y_d , and the control inputs u (equation 2.5).

$$C_U : \Phi(u, Y_d, Y_m). \quad (2.5)$$

Finally, the constraints (C_A) characterize the modulated sources MSf and MSe . These constraints link the control inputs u , and the actuators outputs.

$$C_A = \{C_{MSf}\} \cup \{C_{MSe}\}. \quad (2.6)$$

To make an analogy with the bipartite graph, we define the set of unknown variables as $\{X\}$, and the set of known as $\{K\}$. In the BG sense, the unknown variables $\{X\}$ are the flow and effort pairs associated with power bonds.

$$X = \{e_1(t), f_1(t)\} \cup \{e_2(t), f_2(t)\} \cdots \cup \{e_L(t), f_L(t)\}, \quad (2.7)$$

where L is the total number of power bonds.

Moreover, the set of known variables represents the outer vertices including the flow (Df) and the effort (De) detectors, the flow (Sf) and the effort (Se) sources, the modulated flow (MSf) and effort (MSe) sources.

$$K = \{Df\} \cup \{De\} \cup \{Sf\} \cup \{Se\} \cup \{MSf\} \cup \{MSe\}. \quad (2.8)$$

2.4.2 Causality and bicausality on the BG model

The way in which the unknown variables are calculated depends on the causality assigned to the model. Depending on the type of analysis, two types of causality can be distinguish: The causality, which englobes both *integral* and *derivative* causality, and the *bicausality*. The notion of causality enables to perform structural analysis on the BG model.

2.4.2.1 Causality

In a BG model, the causality is represented by a perpendicular stroke to one end of the bond, meaning that the effort is imposed from the side without the causal stroke to the side with the causal stroke (Figure 2.2). This causality assignment represents the way in which the unknown variable are calculated. Two types of causality can be assigned

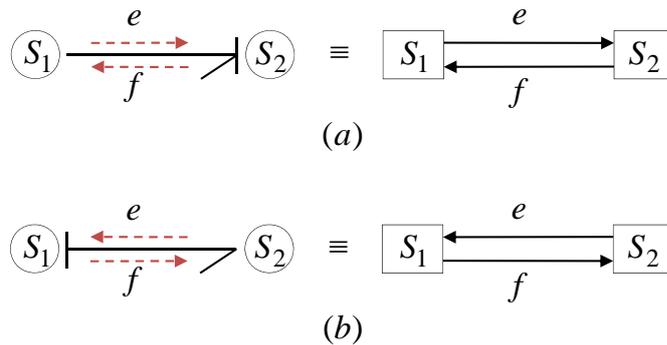


Figure 2.2: Causal BG of the connection between two physical subsystem (S_1, S_2) and corresponding block diagram representation.

to a BG model. For dynamic simulation, a preferred integral causality is assigned to the BG model, meaning that the energy variables are expressed as the integral of the

input (effort or flow). On the other hand, for diagnosis, a derivative causality assignment is preferred. This is because in integral causality, it is necessary to know the system states initial conditions. Thus, rendering the system just-constrained. Nevertheless, in derivative causality the knowledge of the initial conditions is not required and the system becomes over-constrained. A description of the Sequential Causality Assigned Procedure (SCAP) to the bond graph model and further details on the causality assignment are given in Appendix A.

2.4.2.2 Bicausality

The notion of bicausality on the BG models, proposed in [Gawthrop 1995], was initially introduced with the aim of studying inverse system dynamics, parameter and state estimation through the BG model. The concept of causality relies on the idea that physical components can not impose both conjugate power (flow/effort) variables to its connected subsystem. Contrarily to the causality in which both both effort and flow information paths are counter-oriented (dashed arrows). A bicausal assignment extends the concept of causality by also allowing that a subsystem provides both conjugate power variables to its connected subsystem. To represent the bicausality, the causal stroke is divided in two, as depicted in Figure 2.3. One can notice that effort and flow information paths are co-oriented (dashed arrows).

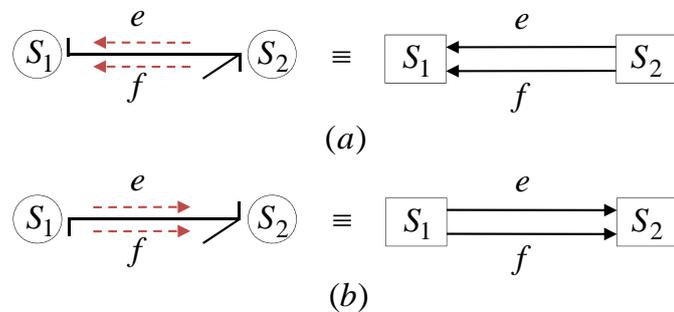


Figure 2.3: Bicausal BG of the connection between two physical subsystem (S_1 , S_2) and corresponding block diagram representation.

2.4.3 Direct Current motor (DC-motor) example

To illustrate the BG modeling procedure together with its set of constraints (C) and variables (Z), let us consider a DC-motor, illustrated in Figure 2.4. The variables and

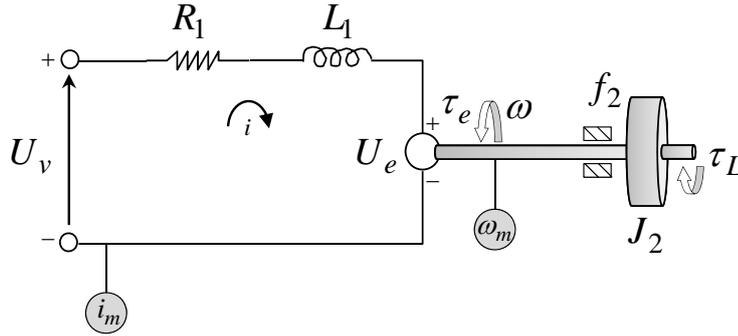


Figure 2.4: Schematic of a DC-motor.

parameters presented in Table 2.1 describe the DC-motor system dynamics.

Table 2.1: Variables of the simplified DC-motor system.

Symbol	Designation	Symbol	Designation
$i(t)$	Motor current	L_1	Inductance of the stator
R_1	Resistance of the stator	$i_m(t)$	Measured current
$U_v(t)$	Input voltage	$U_R(t)$	Resistive voltage
$U_L(t)$	Induced voltage	$U_e(t)$	Back emf
$\tau_e(t)$	Motor torque	J_2	Inertia of rotor and load
f_2	Viscous friction	$\omega_m(t)$	Measured angular velocity
$\omega(t)$	Angular velocity	$\tau_f(t)$	Friction torque
$\tau_J(t)$	Inertial torque	$\tau_L(t)$	Mechanical torque (Load)
k_1	Electromotive force constant		

The word bond graph (WBG) of the DC-motor presented in Figure 2.4, together with the BG model in integral causality is depicted in Figure 2.5. The set of constraints $C = \{C_B\} \cup \{C_S\} \cup \{C_M\} \cup \{C_U\} \cup \{C_A\}$. modeling the DC-motor in integral causality are the following:

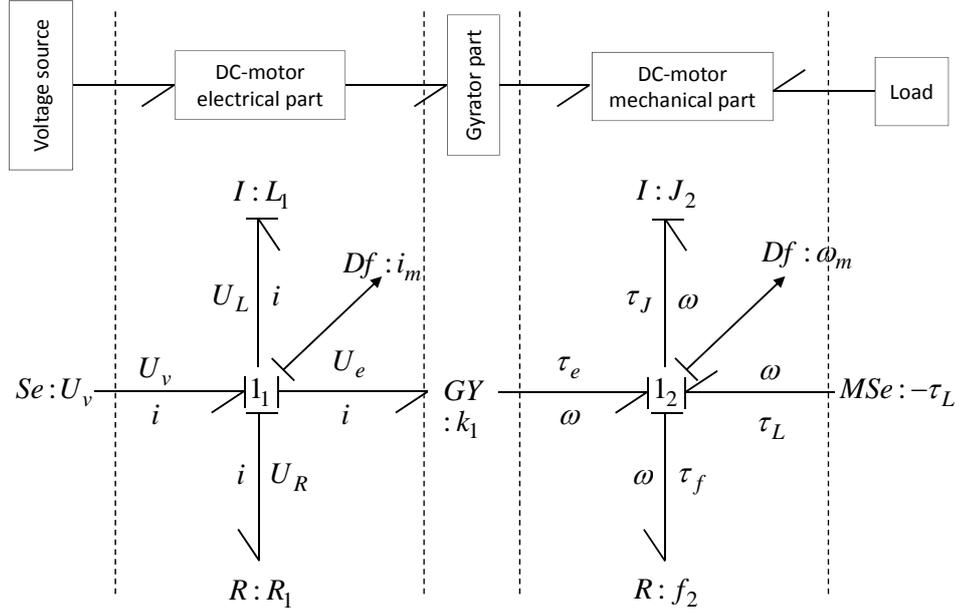


Figure 2.5: WBG and BG model of the DC-motor in integral causality.

$$C_B : \begin{cases} C_{R_1} & : U_R(t) = R_1 i(t) \\ C_{L_1} & : i(t) = \frac{1}{L_1} \int U_L(t) dt \\ C_{f_2} & : \tau_f(t) = f_2 \omega(t), \\ C_{J_2} & : \omega(t) = \frac{1}{J_2} \int \tau_J(t) dt. \end{cases} \quad (2.9) \quad C_M : \begin{cases} C_{i_m} & : i_m(t) = i(t), \\ C_{\omega_m} & : \omega_m(t) = \omega(t). \end{cases} \quad (2.10)$$

$$C_S : \begin{cases} C_{1_1} & : U_L(t) = U_v(t) - U_R(t) - U_e(t), \\ C_{GY1} & : U_e(t) = k_1 \omega(t), \\ C_{1_2} & : \tau_J(t) = -\tau_L(t) - \tau_f(t) + \tau_e(t), \\ C_{GY2} & : \tau_e(t) = k_1 i(t). \end{cases} \quad (2.11)$$

2.4.4 Generation of *ARRs* from BG model

To generate *ARRs*, it is required to obtain the over-determined subsystem ($Card(X) < Card(C)$) of the BG model. An *ARRs* is a constraint calculated from an observable and over-constrained subsystem and expressed in terms of known variables of the system

$f(K) = 0$. Evaluation of an ARR yields a residual (r): $r = Eval[f(K)]$. The initial method for generation of *ARRs* from the BG model was proposed in [Tagina 1996]. In order to obtain the *ARRs* in a systematical manner, [Ould-Bouamama 2003] introduced the causality inversion method procedure. In addition, as proposed in [Sueur 1991], structural observability can be directly concluded from a BG model without the use of any calculations.

Definition 2.7. The system is structurally observable if and only if two conditions are satisfied:

- In a BG model in preferred integral causality, there is a causal path connecting all dynamical elements (I and C) in integral causality to a detector;
- When a preferred derivative causality is assigned to a BG model, all I and C elements must accept a derivative causality. If this is not completely respected, a dualization of the detectors is required to put all I and C elements in derivative causality.

In addition, from a BG model, a system is said to be over-constrained if when the detectors are dualized, all I and C elements accept a derivative causality [Djeziri 2007a].

Detector dualization: In a BG model in preferential integral causality, the sensor is presented by a detector of either effort (De) or flow (Df). This is because the BG model in integral causality furnishes a signal of effort (from a θ -junction) or of flow (from a 1 -junction) for simulation and control. For diagnosis, the measure of the sensor becomes a signal source (of effort SSe , or of flow SSf) imposed onto the observed junction (junction associated to the detector). The transformation of the detectors in signal sources is defined as a *detector dualization*. Figure 2.6 illustrates the dualization procedure. Paths derived using the imposed signal as the starting point produces the method for elimination of unknown variables. In a BG sense, an *ARR*: $f(SSe, SSf, Se, Sf, MSe, MSf, \alpha) = 0$, where α is the parameters vector, and f is a constraint function. The algorithm, called causality inversion approach, for *ARRs*

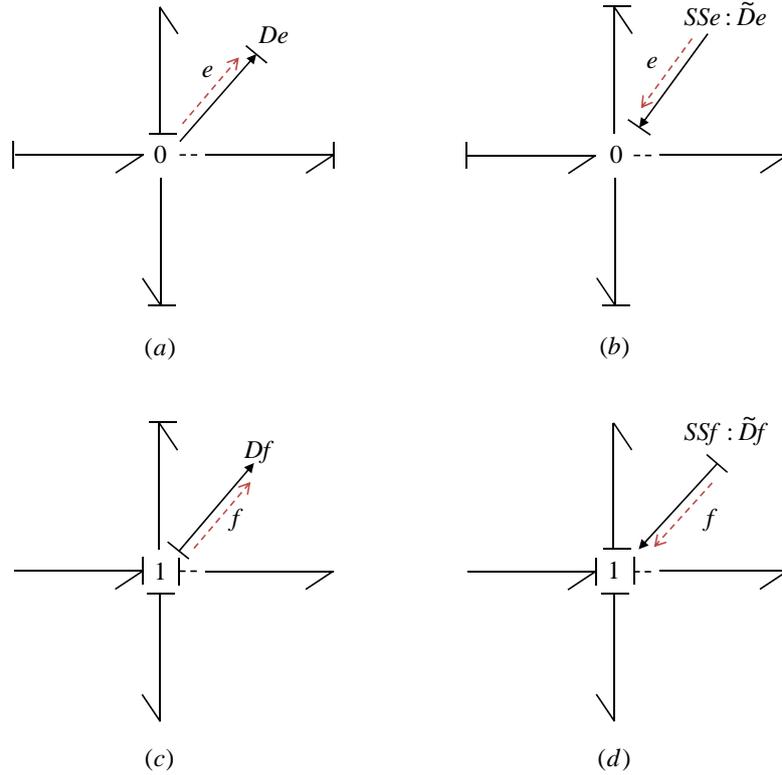


Figure 2.6: (a,c) detector of effort (De), and of flow (Df) and (b,d) signal source of effort (SSe), and of flow (SSf), respectively.

generation from a BG model is synthesized as follows and it enables the computation of *ARRs* in a systematic way [Ould-Bouamama 2003]: **Causality inversion approach for *ARRs* generation from a BG model:**

1. Assign a preferred derivative causality to the BG model and invert the causality of the detectors when possible. Thus, the BG model for diagnosis is obtained.
2. Write the structural constraints (C_S) of the junctions associated with the dualized detectors.
3. For each 0-(1) junction having at least an associated detector:
 - For each junction with an associated detector, an *ARR* is obtained by eliminating the unknown variables. The causal path propagation is used to eliminate the unknown variables.

- When a detector can not be dualized without violating the causality assignment rule, material redundancy may be present in the system. The latter exists if there are direct causal paths from one or more detectors in inverted causality SSf (SSe) to the non inverted one Df (De), without passing through any passive or two-port element [Samantaray 2008b]. This concept is represented in Figure 2.7. In this case, $ARRs$ are equal to the difference between the measures of the redundant sensors.

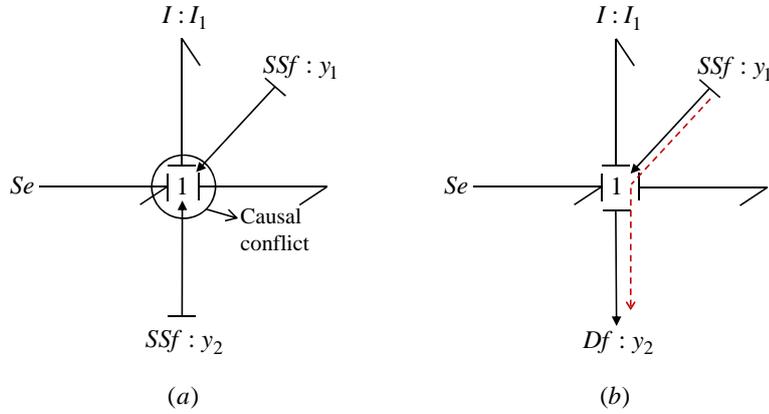


Figure 2.7: Direct detector redundancy in a BG model for diagnosis in preferred derivative causality: (a) Violation of the causality assignment rules if both $SSf : y_1$, and $SSe : y_2$ are dualized, (b) only $SSf : y_1$ is dualized (no violation).

Definition 2.8. In a causal (or bicausal) bond graph representation, a causal path is a series of effort or flow variables successively related according to the model causality assignment.

To illustrate the causality inversion approach with an example, consider the DC-motor in derivative causality (Figure 2.8): Using the algorithm for $ARRs$, the "1₁" junction C_{1_1} is selected:

$$C_{1_1} : U_v(t) - U_R(t) - U_L(t) - U_e(t) = 0, \quad (2.12)$$

where U_R , U_L , and U_e are unknowns that can be eliminated in symbolic format by exploiting the causal properties of the BG model. This is done structurally by covering

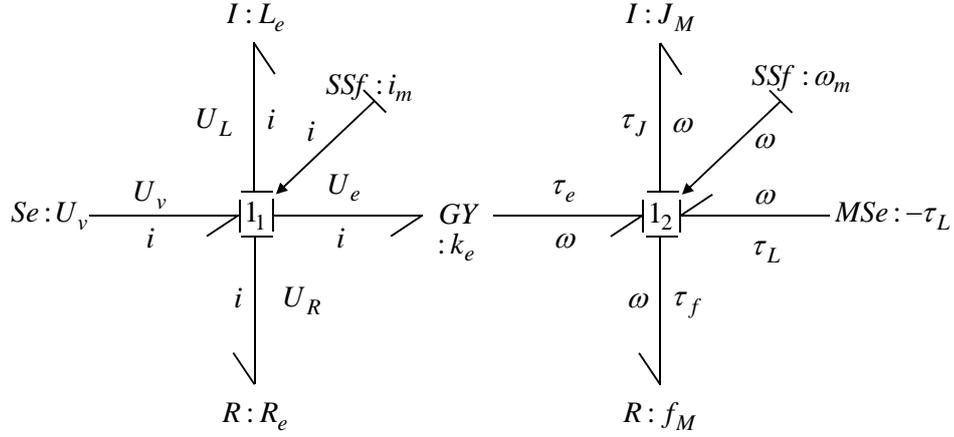


Figure 2.8: BG model of the DC-motor in derivative causality with dualized detectors.

causal paths from unknown to known (detector and source) variables as follows:

$$\begin{aligned}
 U_R &\rightarrow C_{R_1} \rightarrow i \rightarrow C_{i_m} \rightarrow SSf : i_m, \\
 U_L &\rightarrow C_{L_1} \rightarrow i \rightarrow C_{i_m} \rightarrow SSf : i_m \\
 U_e &\rightarrow C_{GY_1} \rightarrow \omega \rightarrow C_{\omega_m} \rightarrow SSf : \omega_m \\
 U_v &\rightarrow : Se : U_v.
 \end{aligned}$$

The *ARR* is obtained by writing its characteristic equations.

$$ARR_1 : U_v(t) - R_1 i_m(t) - L_1 \frac{di_m(t)}{dt} - k_1 \omega_m(t) = 0. \quad (2.13)$$

From the second "1₂" junction C_{1_2} , the second *ARR* is obtained:

$$ARR_2 : -\tau_L(t) - f_2 \omega_m(t), -J_2 \frac{d\omega_m(t)}{dt} + k_1 i_m(t) = 0. \quad (2.14)$$

Once all *ARRs* are obtained, the set of faults that can be structurally monitored (detectable) and isolable must be concluded. To this end, the concept of fault signature is introduced.

2.4.5 Fault isolation

The structures of the residuals form a Boolean fault signature that represents the structural sensitivity of faults on the residuals. Thus, the inclusion of all fault signatures in a fault signature matrix (FSM) allows the knowledge of the faulty components that can be detected, and isolated. Each entry of this FSM (s_{ji}) holds Boolean values, and the fault signature vector (V_{E_j}) of the j^{th} component E_j ($j = 1, \dots, m$) is given by the row vector: $V_{E_j} = [s_{j1}, s_{j2}, \dots, s_{jn}]$, and the values s_{ji} are assigned as follows:

$$s_{ji} = \begin{cases} 1, & \text{if } r_i \text{ is sensitive to a fault in the component } E_j, \\ 0, & \text{otherwise.} \end{cases} \quad (2.15)$$

Where, $i = 1, \dots, n$, and $Card(ARRs) = n$. It means that if $s_{ji} = 1$, a fault in the component E_j influences the residual r_i . In other words, the residual is triggered. On the other hand, if $s_{ji} = 0$, r_i is not sensitive to a fault in the component E_j . These signature vectors are then used to create a FSM, where $FSM \in \mathbb{R}^{m \times (n+2)}$, as illustrated in Table 2.2. This matrix contains all signature vectors, plus two columns that represent fault monitorability ($M_b = [m_{b_1}, m_{b_2}, \dots, m_{b_m}]^T$) and fault isolability ($I_b = [i_{b_1}, i_{b_2}, \dots, i_{b_m}]^T$). m_{b_j} and i_{b_j} are equal to one if a fault in the component E_j

Table 2.2: Fault signature matrix (FSM).

$j \setminus i$	r_1	r_2	\dots	r_n	M_b	I_b
E_1	s_{11}	s_{12}	\dots	s_{1n}	m_{b_1}	i_{b_1}
E_2	s_{21}	s_{22}	\dots	s_{2n}	m_{b_2}	i_{b_2}
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots
E_m	s_{m1}	s_{m2}	\dots	s_{mn}	m_{b_m}	i_{b_m}

is monitorable and isolable, respectively. A component fault E_j is monitorable if at least one s_{ji} of its signature vector V_{E_j} is different from zero ($\exists i_{(i=1, \dots, n)} : s_{ji} \in V_{E_j} \neq 0$). A component fault E_j can be isolated if its signature vector V_{E_j} is different from all others,

$$i_{b_j} = \begin{cases} 1 & \text{if } \forall l_{(l=1, \dots, m)} : V_{E_j} \neq V_{E_l} (j \neq l), \\ 0 & \text{otherwise.} \end{cases} \quad (2.16)$$

Consider the DC-motor example, with the obtained *ARRs* (2.13), (2.14), the FSM presented as Table 2.3 is obtained.

Table 2.3: Fault Signature matrix of the DC-motor from bond graph.

Faults/residual	r_1	r_2	M_b	I_b
$Df : i_m$	1	1	1	0
$Df : \omega_m$	1	1	1	0
$Se : U_v$	1	0	1	0
$R : R_1$	1	0	1	0
$I : L_1$	1	0	1	0
$R : f_2$	0	1	1	0
$I : J_2$	0	1	1	0
$GY : k_1$	1	1	1	0

We remark that the isolability column (I_b) of Table 2.3 is filled with zeros, which means that we are not able to isolate any fault of components E_j .

As it can be concluded from the causality inversion approach for *ARRs* generation procedure, the number of *ARRs* which can be obtained is equal to the number of junctions linked to at least one detector, plus the number of redundant detectors (r) [Samantaray 2008b, Medjaher 2005]. However, this last statement is not always valid. In the next subsection, we will show that it is possible to obtain non-redundant *ARRs* by exploiting detectors combinations.

2.4.6 Generation of additional non-redundant *ARRs*

In Figure 2.9, a general representation of a BG model for diagnosis is illustrated. The model contains one source of effort Se , and two detectors, one of flow ($Df : y_1$), and other of effort ($De : y_2$). As previously explained, for diagnosis purposes, the detectors are transformed into sources of signal of flow ($SSf : y_1$) and effort ($SSe : y_2$), to represent the injection of this information into the system. If the causality inversion approach is used for *ARRs* generation, two *ARRs* of the following form can be obtained

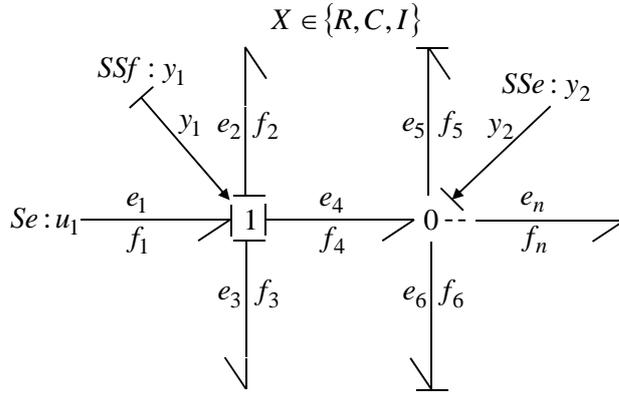


Figure 2.9: General representation of a Bond graph model for diagnosis.

from Figure 2.9:

$$\begin{aligned}
 ARR_1 &: f_1(y_1, y_2, u_1, \dots) = 0, \\
 ARR_2 &: f_2(y_1, y_2, \dots) = 0.
 \end{aligned}
 \tag{2.17}$$

However, this approach does not find the complete set of *ARRs*. Consider that, at a time, only one of the existing detectors are used for diagnosis (as depicted in Figure 2.10 where only *De* : y_2 is used for diagnosis).

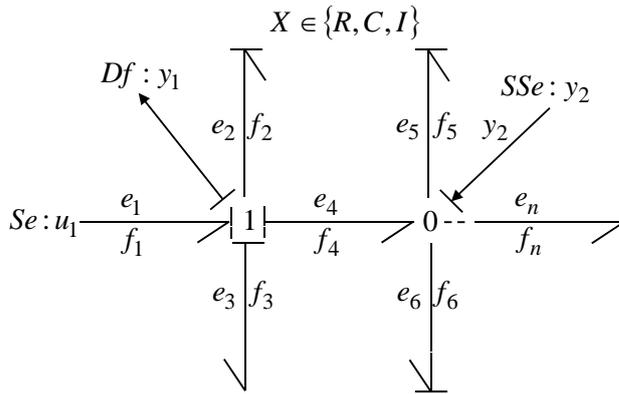


Figure 2.10: General representation of a Bond graph model for diagnosis without dualizing *Df* : y_1 .

In this case, an additional *ARR* of the following form can be obtained:

$$ARR_3 : f_3(y_2, u_1, \dots) = 0.
 \tag{2.18}$$

Again, if only y_1 is used for *ARRs* generation, another *ARR* can be computed:

$$ARR_4 : f_4(y_1, u_1, \dots) = 0. \quad (2.19)$$

It is clear that this technique applies a combinatorial procedure, which causes an increase on its computational complexity. This statement supports the interest of finding structural conditions that can be directly verified on the BG, ensuring the generation of non-redundant *ARRs* by detector combinations. Consider y_k as the k^{th} sensor of the system, where $k = 1, \dots, K$.

Lemma 2.1. *Extra independent *ARRs* can be generated if and only if two conditions are respected :*

1. *The system remains observable and the remaining detectors are dualizable;*
2. *The system is over-constrained with the remaining sensors.*

Proposition 1. *If Lemma 2.1 is satisfied, sensor combinations generate non-redundant *ARRs*, if and only if (a) is respected.*

(a) y_k , is not isolable by the classical causality inversion method, and $ARR_k : f_k(y_k, y_{k+1}, \dots) = 0 \wedge ARR_{k+1} : f_{k+1}(y_k, y_{k+1}, y_{k+2}, \dots) = 0$. Then, it is possible to compute an extra *ARR* (ARR_K), from the junction associated to y_k , so that $ARR_{K+1} : f_K(y_{k+1}, y_{k+2}, \dots) = 0$. In this way, a fault in the sensor y_k can be isolated.

Nevertheless, the combinations referred in Proposition 1 may not allow to generate the additional *ARRs* from the covering path procedure. Therefore, we propose to use the concept of bicausality. To this end, the signal source of flow or effort $SSe/SSf : y_{k+1}$ is replaced by a double source ($S_e S_f$). In this case, $S_e S_f$ represents a source of effort and source of flow, and in addition the non-measured power variable is zero. Finally, a virtual double detector of effort and of flow ($D_e D_f^*$) is also added to the junction associated to y_k . Then, the bicausality is propagated from the double source ($S_e S_f : y_{k+1}$) to $D_f D_e^* : \hat{f}$, as depicted in Figure 2.11, in order to estimate the flow

(\hat{f}). This estimation is then used to compute ARR_{K+1} , which has the same structure as ARR_k .

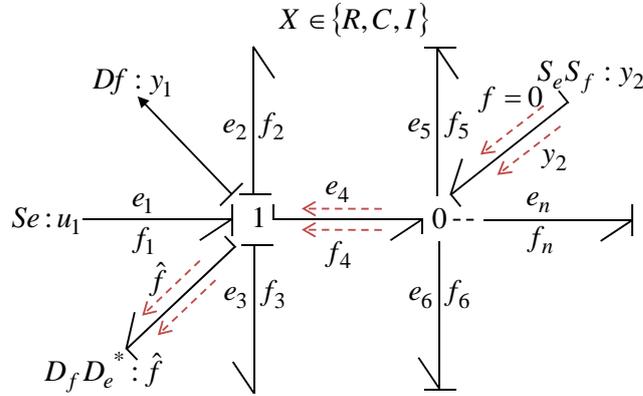


Figure 2.11: General representation of a Bond graph model for diagnosis in bicausality.

Remaining combinations that do not respect Proposition 1 will compute redundant $ARRs$. Thus, there is no interest on its generation. These combinations can increase significantly the isolability of the system. To exemplify this statement consider Figure 2.12 where the detector $Df : i_m$ has been replaced by a double source ($S_f S_e : i_m$) and a virtual double detector ($D_e D_f^* : \hat{\omega}$) is added to the junction 1_2 . From the set of equation (2.20) deduced from the BG, $\hat{\omega}$ is obtained.

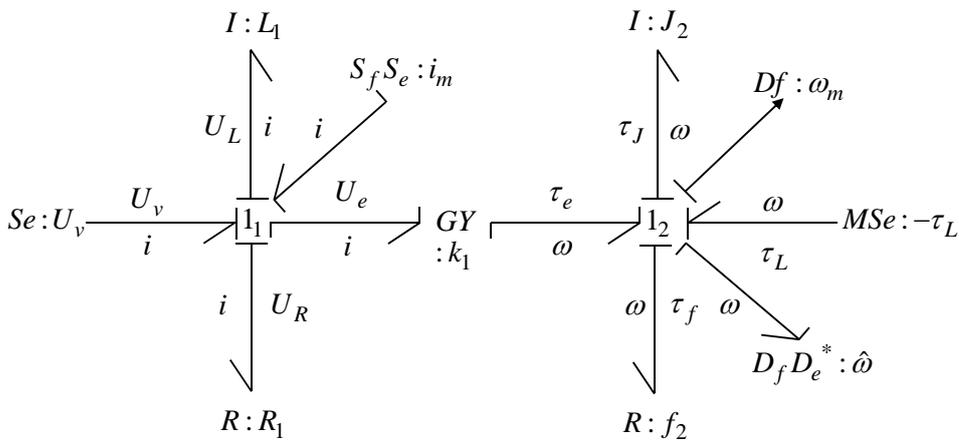


Figure 2.12: Bond graph model of DC-motor in bicausality for angular velocity estimation $\hat{\omega}$.

$$1_1 - \text{junction} : \begin{cases} U_e(t) &= U_v(t) - U_R(t) - U_L(t), \\ i(t) &= i_m(t), \\ U_R(t) &= R_1 i(t), \\ U_L(t) &= L_1 \frac{di(t)}{dt}, \\ \omega(t) &= \frac{U_e(t)}{k_1}, \\ \tau_e(t) &= k_1 i(t). \end{cases} \quad 1_2 - \text{junction} : \begin{cases} \hat{\omega}(t) = \omega(t). \end{cases} \quad (2.20)$$

Finally, if we replace ω_m in (2.14) by $\hat{\omega}$ an additional *ARR* is obtained as follows:

$$ARR_3 : -\tau_L(t) - f_2 \hat{\omega}(t) - J_2 \frac{d\hat{\omega}(t)}{dt} + k_1 i_m(t) = 0. \quad (2.21)$$

The same procedure is applied when $Df : \omega_m$ is replaced by $S_f S_e : \omega_m$, and a virtual double detector ($D_f D_e^* : \hat{i}$) is added to junction 1_1 . In this way, *ARR*₄ is obtained:

$$ARR_4 : U_v(t) - R_1 \hat{i}(t) - L_1 \frac{d\hat{i}(t)}{dt} - \omega_m(t) k_1 = 0. \quad (2.22)$$

Once all *ARRs* are obtained, a new FSM presented as Table 2.4 is obtained.

Table 2.4: Fault Signature matrix of the DC-motor from BG with the additional non-redundant residuals.

Faults/residual	r_1	r_2	r_3	r_4	M_b	I_b
$Df : i_m$	1	1	1	0	1	1
$Df : \omega_m$	1	1	0	1	1	1
$Se : U_v$	1	0	1	1	1	0
$R : R_1$	1	0	1	1	1	0
$I : L_1$	1	0	1	1	1	0
$R : f_2$	0	1	1	1	1	0
$I : J_2$	0	1	1	1	1	0
$GY : k_1$	1	1	1	1	1	1

It can be noticed that the two additional residuals clearly extend the fault isolability properties of the system. In this case, three system components can be isolated $Df : i_m$,

$Df : \omega_m$, and $GY : k_1$.

2.5 Bond graph for structural recoverability analysis

The ultimate objective of using the BG tool is because it enables to couple both structural diagnosis results with control analysis. Therefore, with a single tool, one can model the system, generate *ARRs* for fault diagnosis, and finally study structural recoverability analysis. In this section we propose a structural methodology based on a BG model, which verifies if the system objective (Σ_o) can be achieved in spite of the presence of faults, by considering the complete diagnosis information. In other words, we intend to conclude which faults can be dealt by system reconfiguration and/or fault accommodation, and which faults provoke a system (Σ) shut down.

Sensor faults

Initially, two types of sensors can be distinguished. Namely control sensors and diagnosis sensors. All control sensors are also used for diagnosis. However, the inverse is not always true. For example, in the DC-motor example, $Df : \omega_m$ represents a control sensor, while $Df : i_m$ is only used for diagnosis purposes.

Let us start by considering diagnosis sensor faults. If a fault is presented in one of these sensors, it is required that it is isolable, which can be verified by the FSM of the system, as described in Section 2.4.4, and the re-work of the FDI algorithm should be performed. When a fault occurs in a control sensor, fault isolability is also required. In this case two situations can be distinguished: Either the fault can be estimated (which can be verified structurally by applying a bicausality assignment as detailed in Appendix A.5), or the faulty sensor is switched off (from a graphical point of view its associated outer vertex is removed, thus causing a change in the graphical architecture). In the first case, the estimated information can be used directly to correct the corrupted sensor measurements before the controller uses them. This avoids reconfiguring the

controller and the FDI algorithm should be re-done without considering the faulty sensor. For the second case, a general representation of a BG model is depicted in Figure 2.13-(a). Considering that the detector ($Df : y_1$) is isolable, from a graphical perspective, it can be removed from the model (Figure 2.13-(b)). To illustrate this, the removed detector is represented with a dot-dashed line.

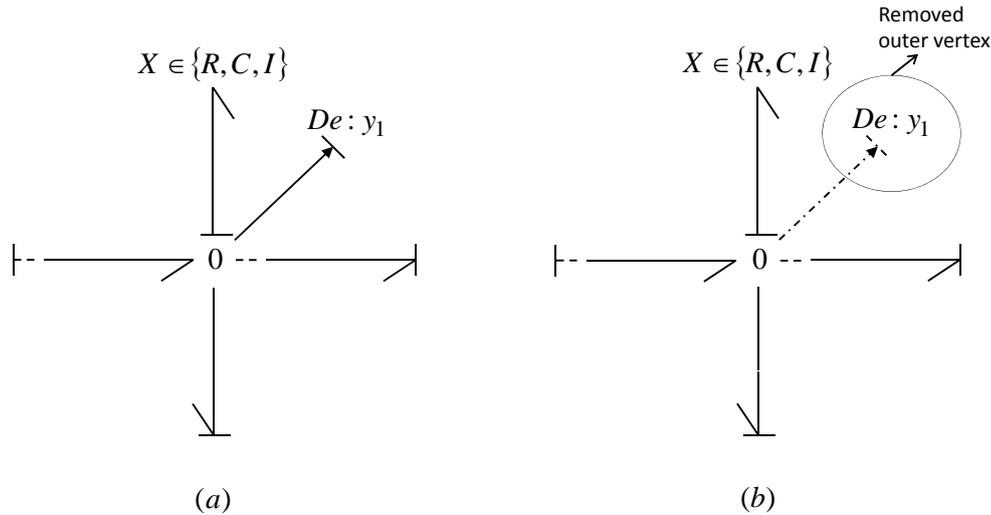


Figure 2.13: General representation of the BG (a) healthy system (b) Faulty system with removed detector outer vertex (faulty sensor).

To synthesize, for such faults, the procedure to verify structural recoverability of the system is the following:

1. Fault must be isolable.
2. Fault presented in diagnosis sensor implies that the FDI must be re-worked.
3. Fault presented in control sensor:
 - (a) Fault is estimable. The reconstructed information can be used directly to correct the corrupted sensor measurement before it is used by the controller.
 - (b) Sensor is removed, two different situations can be considered:
 - i. Material redundancy is available. Therefore, it is enough to use the spare sensor for control purposes. In this case the same control law can still be used.

Structural recoverability analysis from a bond graph model

- ii. Σ' remains structurally observable, with the remaining healthy sensors (analytical measurement redundancy is presented in the system and it can be verified by Theorem 2.7). In this case, it is structurally concluded that system reconfiguration is required.

Independently of the used solution, it is important that the Σ' remains monitorable (note: all faults can be detected). Thus, the FDI algorithm should always be re-worked without considering the faulty sensor.

Actuator faults

Because of the duality between observability and controllability, actuator and control sensor faults are treated in a similar way. Nevertheless, in this case it is related to systems controllability. Structural controllability can be directly concluded from a BG model without the use of any calculations, as proposed in [Sueur 1991].

Definition 2.9. The system is structurally controllable if and only if two conditions are satisfied:

- In a BG model in preferred integral causality, there is a causal path connecting a control source to each I, and C element in integral causality;
- When a preferred derivative causality is assigned to a BG model, all I and C elements must accept a derivative causality. If this is not completely respected, a dualization of the control sources is required to put all I and C elements in derivative causality.

Again, the FDI procedure should be able to isolate the fault. Then, it can be verified structurally if the fault is estimable (Appendix A.5) and if the fault is not severe fault accommodation can be performed. Otherwise, direct or analytical input redundancy has to be available in the system. Moreover, in this case the faulty actuator is not used for control purposes, and it does not furnish power to Σ' (from a graphical point of view its associated outer vertex is removed, thus causing a change on the graphical architecture). A general representation of a BG model is depicted in Figure 2.14-*(a)*. Considering that the actuator ($Sf : u_1$) is isolable, from a graphical perspective, it can be removed from the model (Figure 2.14-*(b)*). To illustrate this, the removed actuator is represented with a dot-dashed line. Extending the results of direct sensor redundancy (presented in Section 2.4.4), direct input redundancy is presented in a BG model by two or more sources if the following proposition holds.

Proposition 2. *Material input redundancy is presented in the system if the shortest causal paths linking two or more sources to a detector meet the same passive and two-port elements.*

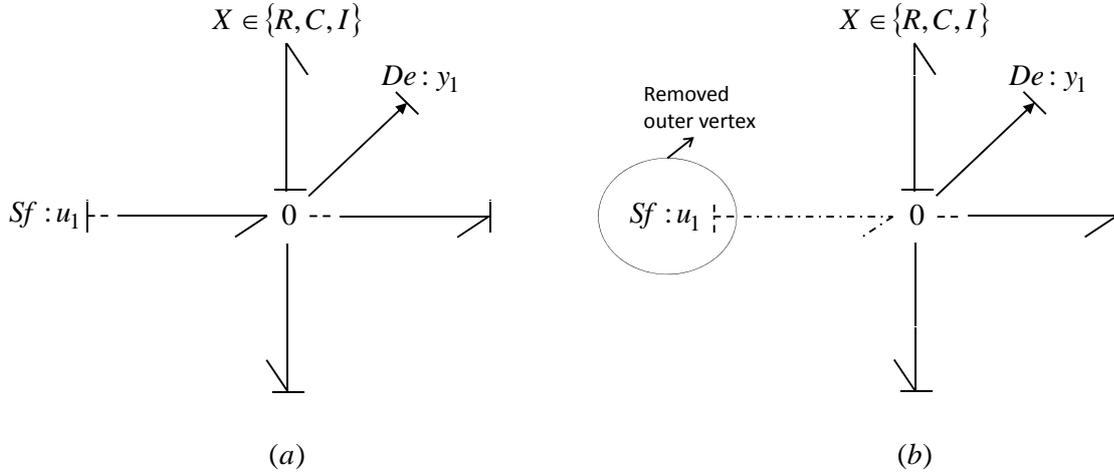


Figure 2.14: General representation of the BG (a) healthy system (b) faulty system with removed actuator outer vertex (faulty actuator).

Definition 2.10. The shortest causal path from a source (Se (MSe)/ Sf (MSf)) to a detector (De/Df) is the one involving the minimal number of energy storage elements when following the path from Se (MSe)/ Sf (MSf) to De/Df .

Hypothesis 2.1. The dynamics of the control sources can always be turned off so that they do not affect the system.

In Figure 2.15 an example of material input redundancy is given. Its shortest causal paths between each source to a detector are described in (2.23) for sources of flow (Sf), and in (2.24) for sources of effort (Se), corresponding to part (a), and (b) of Figure 2.15, respectively.

$$(a) \begin{cases} Se_1 \rightarrow I : I_1 \rightarrow Df_1, \\ Se_2 \rightarrow I : I_1 \rightarrow Df_1. \end{cases} \quad (2.23)$$

$$(b) \begin{cases} Sf_1 \rightarrow C : C_1 \rightarrow De_1, \\ Sf_2 \rightarrow C : C_1 \rightarrow De_1. \end{cases} \quad (2.24)$$

It is clear that both paths illustrated in (2.23) involve the same elements, meaning

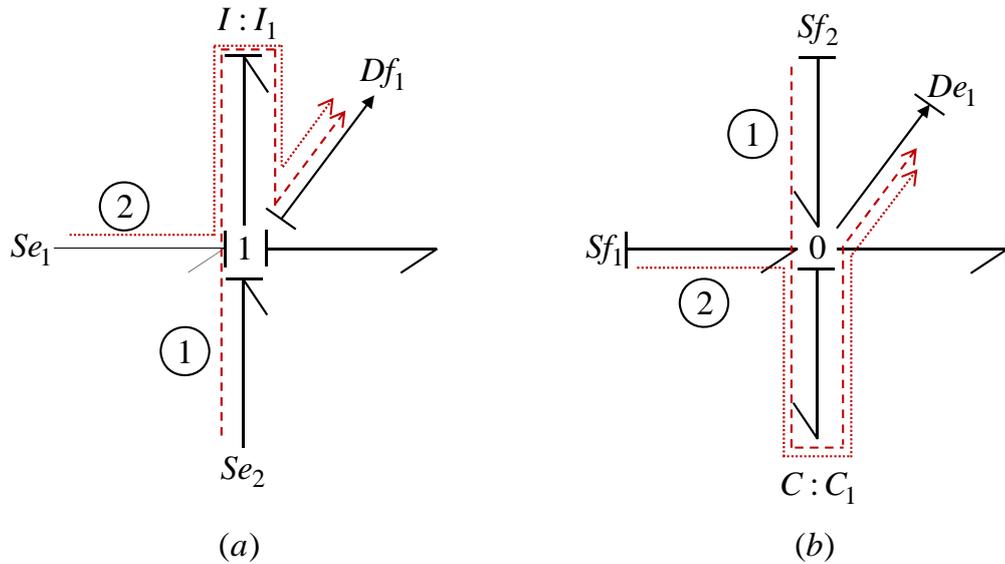


Figure 2.15: Material input redundancy: Direct causal path from effort sources (S_e), part (a) (flow sources (S_f), part (b)) to sensor involving the same elements.

that both inputs affect the output exactly in the same way (existence of material input redundancy). Thus, Se_1 , and Se_2 fall in this type of redundancy. Likewise, the same can be concluded about Sf_1 , and Sf_2 .

To synthesize, for such faults, the procedure to verify structural recoverability of the system is the following:

1. Fault must be isolable.
2. The input fault can be estimated and it is not severe, in this case fault accommodation can be selected,
3. Primary actuator is turned off, two different situations can be considered:
 - (a) Material redundancy is available, such situation enable the use of the same control law by turning on spare actuator.
 - (b) Σ' remains structurally controllable, with the remaining healthy actuators. In this case, it is structurally concluded that the system reconfiguration is required.

Remark. The level of the fault severity can not be evaluated from structural analysis. However, it is used here in order to enable distinguish the selection between fault accommodation and system reconfiguration.

Plant faults

The main difference between plant and sensor/actuator faults is that keeping structural observability/controllability is not a sufficient condition to ensure that Σ^f can remain in process. This is stated because it is not possible to remove the dynamics of the faulty component from the controlled system, contrarily to sensors and actuators that can be switched off.

To accomplish fault accommodation, a bicausal assignment [Touati 2012] has to be performed. Since the fault is estimated, the structure of $\Sigma^{\hat{f}}$ is equal to the structure of the initial system, where structural controllability, observability and monitorability do not change. Then, fault accommodation can be applied to compensate the fault effects. If the fault can not be estimated or it is too severe, the procedure presented below for non-isolable faults has to be followed.

Non-isolable faults

Often, when a fault occurs in a process system, the FDI algorithm is not able to indicate the exact fault location, i.e., a finite subset of possible faulty components have the same fault signature. In this case, fault accommodation can not be performed. However, system reconfiguration may be possible if all the dynamics of the finite subset of possible faulty components can be removed from Σ_o . This is because a part of Σ^f becomes unknown. However, to remove the dynamics of the fault from Σ_o , a cut of the power transfer between the faulty subsystem to the healthy ones is necessary. This is called *Path Breaking* (PB).

Definition 2.11. A PB is presented in a system if and only if it is possible to cut the power propagation between a subsystem to Σ_o . In other words, the dynamics of a subsystem can be prevented from affecting the system objectives.

From the physical structure of BG models, it is possible to study the causal path propagation, between components to Σ_o . PBs are not explicitly represented in BG models, hence let us start by extending the BG methodology to represent them. Recalling that bonds represent the power flow between system components or subsystems. Thus, in a BG sense a PB occurs when the propagation of power ($P = effort \cdot flow$) through the bond connecting two subsystems is equal to zero. This may be caused by using some controlled elements, such as: R , I , C , MGY , MTF , Se , MSe , Sf , or MSf . Note that due to physical constraints of these controlled elements, the location of PBs have to be indicated by human experts at the system design stage (offline during model design), and a label is added to it. In this label the floating value (z) is included, where z is chosen by evaluating the behavioral/structural constraints of the controlled BG element so that $P_1 = e_1 \cdot f_1$, and $P_2 = e_2 \cdot f_2$ are equal to zero ($\exists z \in \mathfrak{R}^+ : P_1(z) = 0 \wedge P_2(z) = 0$), where $e_{1,z} = \Phi_z(f_{1,z})$, thus causing a PB. P_1 is the power that propagates to the controlled element that causes the PB, and P_2 is the power that flows from the faulty subsystem to the healthy one. To clearly understand the concept of PB and its physical concept, consider Figure 2.16-(a, b, c), where three controlled elements that can cause a PB are illustrated. S_{1F} and S_2 are a faulty and a healthy subsystem, respectively. In part (a), the PB is provoked if $k = 0$. This can be easily concluded from the structural equations, and their developments presented in (2.25).

$$\left\{ \begin{array}{l} e_1 = k \cdot f_2, \quad e_2 = k \cdot f_1, \\ \text{if} \quad z = k = 0. \\ \text{Then,} \quad e_1 = 0 \cdot f_2 = 0, \quad e_2 = 0 \cdot f_1 = 0, \\ \quad \quad P_1 = e_1 \cdot f_1 = 0, \quad P_2 = e_2 \cdot f_2 = 0. \end{array} \right. \quad (2.25)$$

As concluded from the structural equations, and their developments presented in

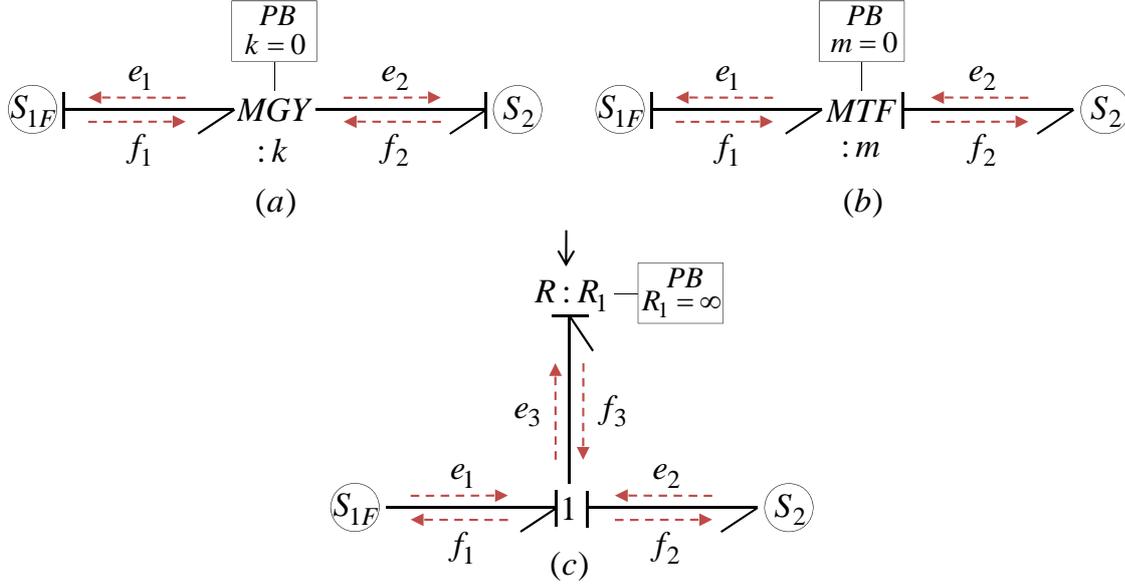


Figure 2.16: Some controlled BG elements that can cause a PB: part (a) Controlled gyrator ($MGY : k$), (b) Controlled transformer ($MTF : m$), and (c) Controlled resistive element ($R : R_1$).

(2.26), a PB is caused in part (b) if $m = 0$.

$$\left\{ \begin{array}{l} e_1 = m.e_2, \quad f_2 = m.f_1, \\ \text{if } z = m = 0. \\ \text{Then, } e_1 = 0.e_2 = 0, \quad f_2 = 0.f_1 = 0, \\ P_1 = e_1.f_1 = 0, \quad P_2 = e_2.f_2 = 0. \end{array} \right. \quad (2.26)$$

Finally, in part (c) the PB can be concluded from the behavioral equations 2.27:

$$\left\{ \begin{array}{l} f_3 = \frac{f_3}{R_3}, \quad \text{if, } z = R_1 = \infty. \\ \text{Then, } f_1 = f_2 = f_3 = 0, \\ P_1 = e_1.f_1 = 0, \quad P_2 = e_2.f_2 = 0. \end{array} \right. \quad (2.27)$$

In addition, if a PB is caused by a modulated R , I , or C with an associated 1-(0) junction, its common flow (effort) variable is set to zero by the controlled element. If it is caused by a MGY , MTF , Se , or Sf , the imposed outputs of these elements are equal to zero.

Hypothesis 2.2. The elements used to provoke a PB can always be set to their necessary values (z), even if this element is subject to a fault.

Taking into consideration the information contained in FSM, let us define $F_{set} = \{F_{sig}^b | b \in (1 \dots B)\}$, where F_{sig}^b contains the set of components (E) with the same fault signature and B the total number of different signatures. For the DC-motor example, there are five fault signatures, hence $B=5$ and F_{sig}^b is defined as follows.

$$\left\{ \begin{array}{l} F_{sig}^1 = \{Df : i_m\}, \\ F_{sig}^2 = \{Df : \dot{\theta}_{ej}\}, \\ F_{sig}^3 = \{R : R_1, I : L_1, Se : U_v\}, \\ F_{sig}^4 = \{R : R_2, I : J_2\}, \\ F_{sig}^5 = \{GY : k_1\}. \end{array} \right. \quad (2.28)$$

Depending on the triggered signature (b) of the fault, the necessary and sufficient controlled BG elements used to cause the PB can be obtained from the following conditions.

PB conditions:

Condition 1: Controlled elements (R , I , C , MGY , or MTF) for which all the causal paths from the components (E) belonging to F_{sig}^b ($E \in F_{sig}^b$), must pass by the component that causes the PB or by its associated junction before it achieves the Σ_o .

Condition 2: All the causal paths from sources (Se , MSe , Sf or MSf) that pass by the components belonging to F_{sig}^b or by their associated junctions before achieving the Σ_o must be stopped.

From a graphical point of view, the outer vertices that do not affect anymore Σ_o are removed, thus causing a change on the graphical architecture which generates Σ' . In this case, the dimension of the system states, sources, and detectors may decrease. Thus, controllability/observability must be verified for Σ' . Finally, the FDI algorithm

Structural recoverability analysis from a bond graph model

has to be performed to Σ' in order to verify if the monitorability conditions remain respected.

To synthesize, for such faults, the procedure to verify structural recoverability of the system is the following.

1. Obtain the triggered fault signature F_{sig}^b .
2. Verify the existence of a PB.
3. Σ' remains structurally controllable, observable, and monitorable. In this case, it is structurally concluded that system reconfiguration is required.

Note that if controllability, and/or observability are lost due to the presence of an actuator or sensor fault, the existence of a PB can also be a valid technique to recover controllability, and/or observability of Σ' .

To represent the described procedure for structural fault recoverability analysis in an automatic manner, the algorithm illustrated in Figure 2.17 is proposed. Even if, the algorithm is quite self-explanatory, some detailed information may be added. The algorithm procedure is activated as soon as a fault is detected in the system, and it receives the information of the FDI algorithm. Then, this information is exploited to answer some questions in order to conclude structurally if fault accommodation, or system reconfiguration can be performed. Moreover, if none of the previous strategies can be successfully achieved, the fault can not be recovered. This algorithm is further detailed in chapter 4 with the add of an example.

Note that the actual achievement of Σ_o with stability and acceptable performance can not be ensured structurally. This may be verified by the implementation of a control strategy, which is beyond the scope of the present work. Also, a Σ may remain able to achieve its Σ_o if it has an uncontrollable/unobservable part that remains stable, again this can not be structurally verified.

Finally, because the diagnosis information is taken into consideration, this algorithm is not applied for each possible fault but for each F_{sig}^b . Hence, by performing the latter

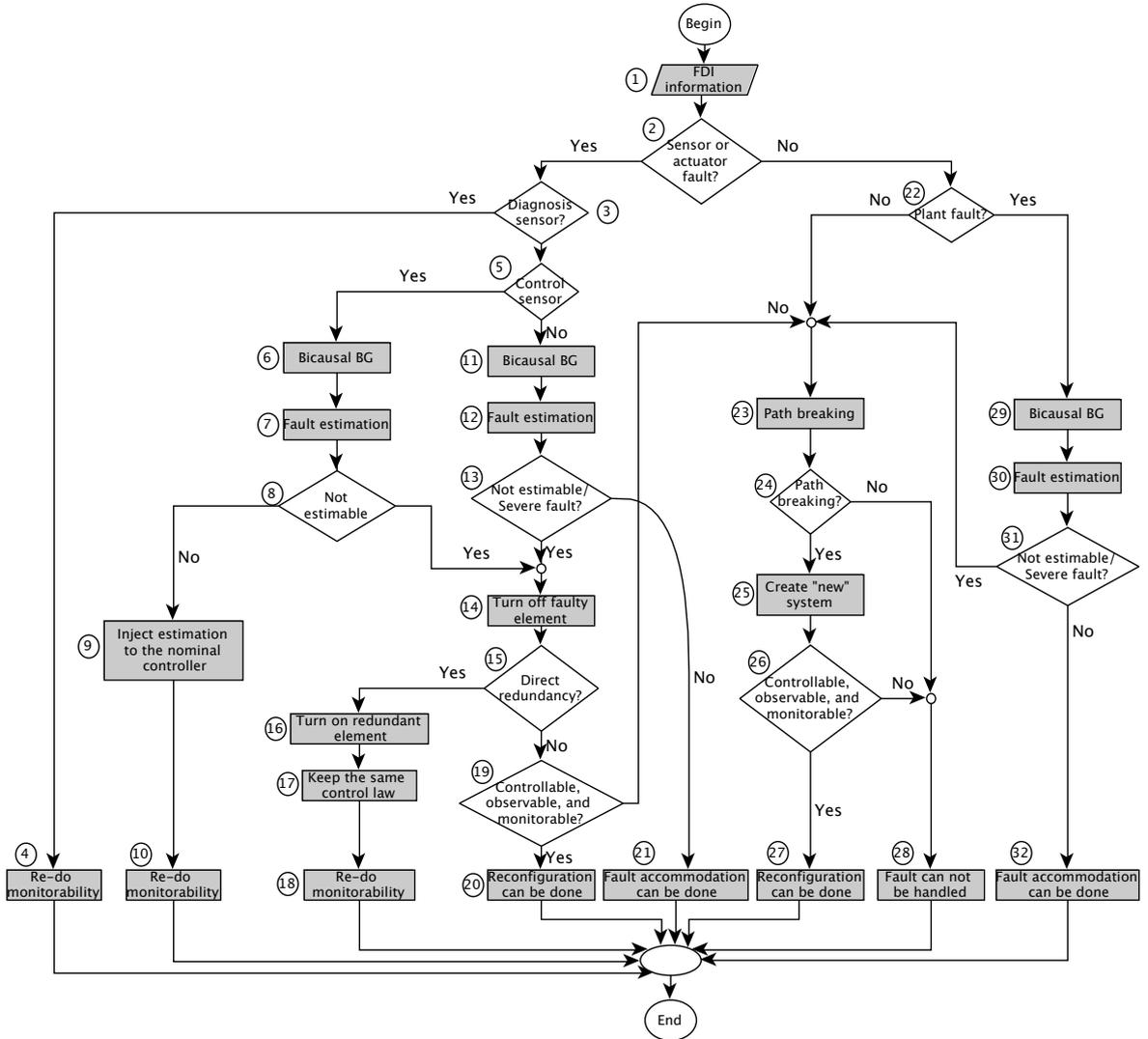


Figure 2.17: Algorithm of structural fault recoverability analysis.

the following Boolean Fault Signature and Recoverability Matrix (FSRM) is obtained:

Table 2.5: Fault Signature and Recoverability Matrix (FSRM).

	r_1	r_2	\dots	r_n	M_b	I_b	SR	SA
E_1	s_{11}	s_{12}	\dots	s_{1n}	m_{b_1}	i_{b_1}	sr_1	sa_1
E_2	s_{21}	s_{22}	\dots	s_{2n}	m_{b_2}	i_{b_2}	sr_2	sa_2
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots	\vdots	\vdots
E_m	s_{m1}	s_{m2}	\dots	s_{mn}	m_{b_m}	i_{b_m}	sr_m	sa_m

Where, SR stands for structural reconfigurability, and SA for structural accommod-

ability. They are binary columns that are filled as follows:

$$SR = \begin{cases} 1 & \text{if when the fault signature } F_{sig}^b \\ & \text{is triggered then the } \Sigma \text{ can be reconfigured,} \\ 0 & \text{otherwise.} \end{cases} \quad (2.29)$$

$$(2.30)$$

$$SA = \begin{cases} 1 & \text{if when the fault signature } F_{sig}^b \\ & \text{is triggered then the } \Sigma \text{ can be accommodated,} \\ 0 & \text{otherwise.} \end{cases} \quad (2.31)$$

From Table 2.5, one can define ($CF_{sig} = \{F_{sig}^b : SR_b = 0 \wedge SA_b = 0\}$) as the set of critical fault signatures and ($CF_{ele} = \{E_j\} \in \{CF_{sig}\}$) as the set of critical faulty elements.

2.6 Conclusions

In this chapter we developed a procedure to determine the structural recoverability of the system when subject to faults, in the absence of complex calculations. We remark that in some situations, fault isolability is not a necessary condition for system recoverability. Due to this fact, the number of faults that a system can tolerate when proper control actions are applied may increase. As a limitation of this work, it can be stated that the required energy to control and observe the system is not known because of the use of a structural approach that is independent of the system numerical values. The proposed algorithm is a preliminary study before implementing a closed loop control. It allows from the systems design stage, to study all conditions related to fault recoverability, and to obtain the set of critical and non-critical faults, inspired from the graphical and structural properties of the BG model. This extension work of the BG

model-based structural analysis for fault recoverability can be exploited for synthesis, under certain conditions, of an adaptive compensation of the fault effect. For this, the disturbing power delivered by the fault is estimated. This is demonstrated in the next Chapter 3. Finally, to detail the proposed structural approach it will be applied on a Heavy-Sized Intelligent Autonomous Vehicle (IAV) in Chapter 4.

Chapter 3

Local adaptive fault compensation

3.1 Introduction

As presented in the previous chapter, the fault isolation and estimation is a main requirement for fault compensation through fault accommodation strategies. In this chapter we propose a way to compensate the faults that can be estimated (concluded from the FSRM), directly from the inverse BG-LFT¹.

In addition, the AFTC community usually consider the outputs of the FD algorithms as information that furnishes the location of the fault and its magnitude. Hence, we also propose a novel way to consider the fault information. In fact, the fault in a system can be considered either as information or as modified power, with respect to the nominal conditions. This idea comes from the fact that a fault in a physical component modifies the power exchanged between components of the system.

Since the BG model explicitly displays these exchanges, it is relatively easy to capture this modified power. In addition, it can then be introduced into model of the system by using additional modulated sources of effort (*MSe*) or flow (*MSf*). In this way, dynamic models of the faulty system are obtained. To capture the power generated by the fault, the fault estimation algorithm proposed by [Touati 2012] is exploited. The referred procedure is performed for fault estimation, nevertheless if the fault is not isolable, we employ the same strategy not to estimate the fault but to estimate the modified power caused by the unknown fault. **In this chapter, we demonstrate how the BG model-based structural analysis for recoverability can help for the synthesis of an adaptive compensation that considers the induced power caused by the fault.** In addition, we also extend the results with respect to classical conditions of fault recoverability. To validate these results, we propose an inverse control strategy that is easily obtained from the BG model. The idea is to model the inverse system together with the power generated by the fault in order to compute suitable control actions that compensate the fault.

The proposed procedure is initially exploited for structural analysis as follows:

¹BG-LFT - Bond Graph in Linear Fractional Transformation (detailed in Appendix A)

- Obtain the set of different fault signatures (F_{set});
 1. If the fault is isolable, model the fault using the BG-LFT;
 2. If the fault is not isolable, model the fault using the BG-LFT in one of the elements belonging to the fault signature;
- Apply the procedure of fault estimation;
- If the fault can be structurally estimated, the set of possible faults can be structurally accommodated.

In this chapter, we will illustrate in more detail the proposed procedure.

Hypothesis 3.1. The development of this work is done under the assumption of system inversion.

3.2 Fault modeling

A forward, also called direct model of a physical system is used to predict the outputs of a system to given inputs under the assumption that faults are not presented in the system. Hence, if an accurate modeling is performed, the output vector of the physical system (y_m) is approximately the same as the output vector predicted by the model (y_{pr}), ($e = y_m - y_{pr} \approx 0$) (Figure 3.1-(a)).

As stated by [Blanke 2003], a fault is a deviation of the system structure or parameters from their standard conditions. Which, from a physical point of view, involves an unknown modification of the physical parameters, inputs or outputs of the system. Hence, if the physical system is subject to faulty conditions, the direct model of the system becomes invalid, causing the predicted outputs (y_{pr}) to diverge from the measured ones (y_m) ($e = y_m - y_{pr} \neq 0$) (Figure 3.1-(b)).

We recall that the main objective of fault accommodation is to compensate the fault by identifying the model of the faulty system. Hence, the fault must be isolated and estimated, and once the parameter of the faulty value is known, a model of the

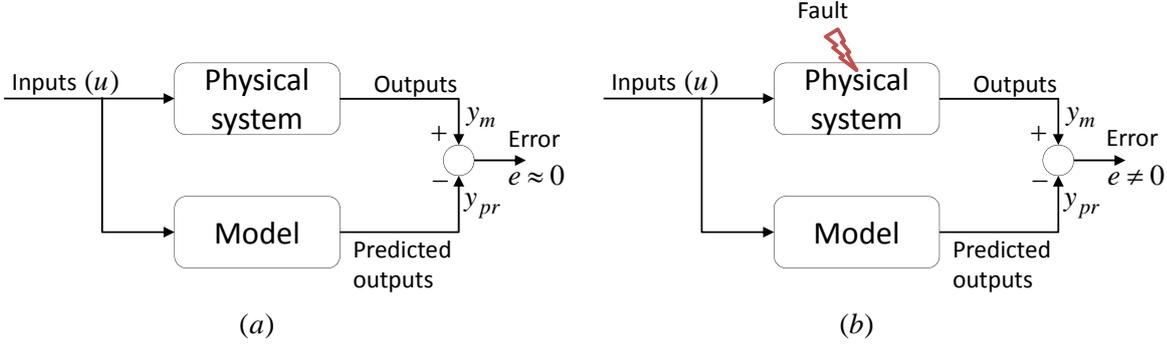


Figure 3.1: Comparison between the system outputs and its forward model in fault free case (a), and in faulty situation of the system (b).

faulty system is obtained. Nevertheless, since the fault isolation is difficult to achieve for a large set of physical components, there are often small sets of faults for which fault accommodation is performed.

Although, since the BG model relies on the representation of the exchanged power between the system physical components, this concept of power propagation can be helpful for fault compensation. To enable this, the faults are classified in two groups, depending on the way in which they affect the system. The first group assembles input sources and plant faults while the latter one is for sensor faults.

Input and plant faults are in the same group because they affect the system in a similar way. Indeed, the occurrence of one of these faults modify the power exchanged in the system, with respect to the nominal situation. In a BG model, this modification of power is captured by a variation of the power variables (e/f). This is easily understandable from a BG model (Figure 3.2). Recalling that in a BG model, the connection between different elements of the system is done through the θ - and 1 - junctions that model the concept of power conservation (3.1):

$$\begin{aligned}
 1 - \text{junction: } \sum e_i(t) &= 0, \\
 0 - \text{junction: } \sum f_i(t) &= 0.
 \end{aligned} \tag{3.1}$$

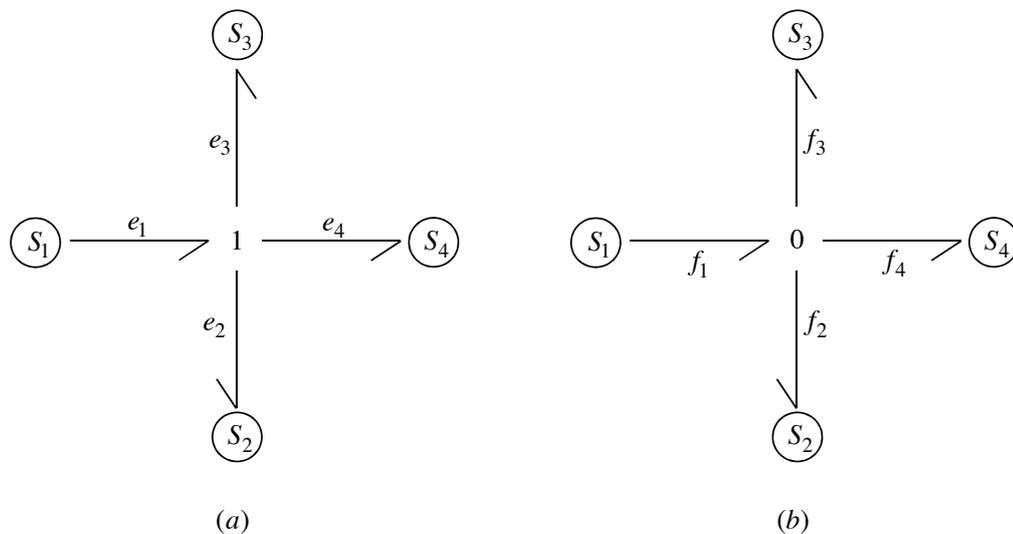


Figure 3.2: General acausal BG model.

where, $i = 1, \dots, n$ and n is the number of bonds connected to the junction. P_i represents the power associated to the i^{th} junction.

Therefore, we can consider that the occurrence of a fault in one of the sources or passive elements associated to a $1-(0-)$ junction, modifies the power in this junction with respect to nominal case. Hence, the power exchanged in the physical system is different from the one in BG model. Finally, to obtain a valid model the difference of exchanged power is captured and injected into the system with a modulated source to the junction associated to the faulty element, as depicted in Figure 3.3.

From this BG, equation (3.2) is obtained.

$$\begin{aligned}
 1 - \text{junction: } \sum e_i(t) - e_F(t) &= 0, \\
 0 - \text{junction: } \sum f_i(t) - f_F(t) &= 0.
 \end{aligned} \tag{3.2}$$

We remark that this concept is not valid for sensor faults. Actually, sensor faults trigger the residuals not because an additional power is presented in the system but because one of the information furnished by (SSf/SSe) is inaccurate. Hence, causing an incoherence between the different known variables used to evaluate the validity of

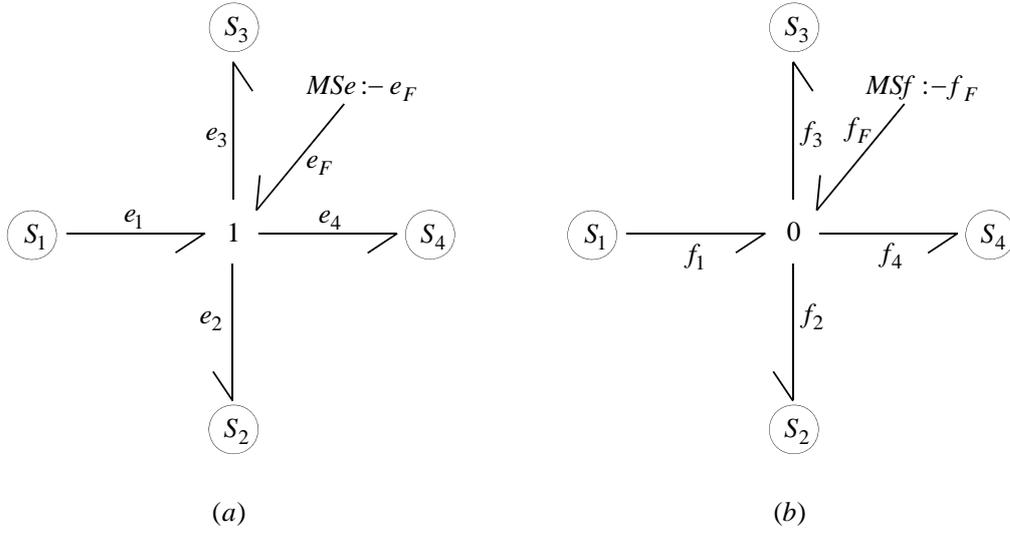


Figure 3.3: Acausal BG model with an additional source imposing the power generated by the fault. In part (a) *1-junction*, the power is represented as a modulated source of effort $MS_e : -e_F$, and in part (b) *0-junction*, the power is represented as a modulated source of flow $MS_f : -f_F$.

the power conservation ($\text{eval}(ARRs)$).

For clarification purposes, let us consider a general representation of a BG model for diagnosis presented in Figure 3.4-(a). As aforementioned, an $ARR : e_1 - e_2 - e_3 - e_4 = 0$ can be obtained, and let us consider that e_1 and e_4 are known variables. Equations (3.3) are used to calculate e_2 and e_3 , where S_2 and S_3 are the transfer function of their corresponding subsystems. Hence, if a fault is presented in S_2 (respectively, S_3), the actual effort e_2^s (respectively, e_3^s), presented in the physical system changes with respect to the normal conditions, e_2 and e_3 that are obtained from the bond graph model (3.3).

$$\begin{aligned} e_2 &= f(S_2, y), \\ e_3 &= f(S_3, y). \end{aligned} \tag{3.3}$$

However, if a fault is presented in a sensor (for example, an additive faulty effect (y_F) with respect to the nominal measure (y), as in Figure 3.4-(b)), then the estimations of both efforts will be corrupted (e_2^c, e_3^c) (equation 3.4). On the other hand, the efforts in

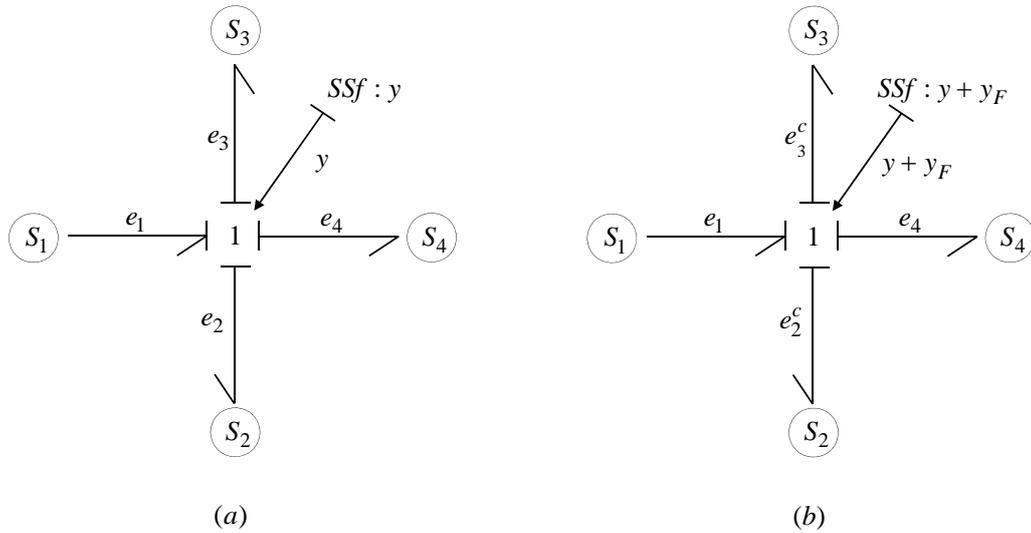


Figure 3.4: General BG model for diagnosis with healthy sensor (a) and with faulty sensor (b).

the physical system e_2^s and e_3^s are the same as in normal conditions. In this case, it is the efforts obtained from the BG model for diagnosis that are "faulty" (computational information) instead of the one in the physical system.

$$\begin{aligned} e_2^c &= f(S_2, y + y_F), \\ e_3^c &= f(S_3, y + y_F). \end{aligned} \quad (3.4)$$

Therefore, if in the occurrence of a plant or an input fault, one is able to acquire the power generated by the fault and suitably feed it into the system, a model of the faulty system is identified. Nevertheless, in the presence of a sensor fault it can not be done because there is no actual fault in the physical system.

To model a fault in a BG model, [Touati 2012] proposed a procedure that models faults elegantly in a BG model by applying the BG-LFT procedure, and exploits it for fault estimation, as described in Appendix A.

For illustration purposes on the fault modeling procedure, let us consider a fault in the GY element of the DC-motor detailed in Section 2.4, and as explained, this fault is

Local adaptive fault compensation

isolable. By applying the BG-LFT formalism for fault modeling procedure, Figure 3.5 is obtained. The output z (which is the flow in the junctions associated to the faulty GY) is represented by a virtual sensor Df^* . F_{GY} represents the faulty value with respect to the nominal value of the gyrator parameter (k_n), and MSf is the modulated source introducing the power caused by the fault (W_a/W_b) into the system.

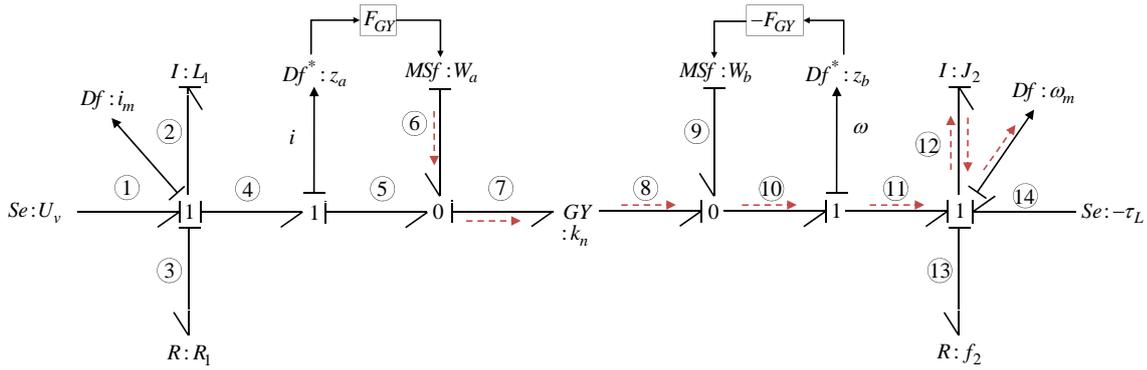


Figure 3.5: BG model of the DC-motor in preferred integral causality, with a fault in the GY element.

The equations developed from the bond graph model show that to determine both e_4 (3.5) and e_{11} (3.6) the effect of the fault ($F_{GY}.\omega$ and $F_{GY}.i$) is considered.

$$\begin{aligned} \omega &= f_{13} = f_{12} = f_{11} = f_{10}, & f_8 &= -f_9 + f_{10}, & W_b &= -F_{GY}.\omega \\ f_8 &= -W_b + \omega, & e_4 &= e_7 = e_6 = e_5 = f_8.k_n, \\ e_4 &= (F_{GY}.\omega + \omega)k_n. \end{aligned} \quad (3.5)$$

$$\begin{aligned} i &= f_1 = f_2 = f_3 = f_4 = f_5, & f_7 &= f_5 + f_6, & W_a &= F_{GY}.i \\ f_7 &= W_a + i, & e_8 &= e_9 = e_{10} = e_{11} = f_7.k_n, \\ e_{11} &= (F_{GY}.i + i)k_n. \end{aligned} \quad (3.6)$$

Once the fault is modeled, a bicausal procedure is performed for fault estimation. In this case, a preferred derivative causality is assigned to the bond graph model. Then, the modulated sources of flow modeling the fault are replaced by a double detector

($DfDe$). In addition, the detector with the shortest causal path between a sensor and the modulated source is replaced by double sources ($SfSe$). Finally, a bicausal shortest path is assigned between the double source $SfSe$ to the double detector $DfDe$ (Figure 3.5 (red dashed arrows)). The result of this procedure is illustrated in Figure 3.6.

Definition 3.1. (Causal path length)[Rahmani 1996]

- In a BG model with all storage elements in integral causality, the length l of a causal path from a set of element $S_1 = \{Se, Sf, I, C, R\}$ to $S_2 = \{De, Df, I, C, R\}$ is equal to the number of storage (I/C) elements covered +1 if the end of the path is an I/C element.
- In a BG model in preferred integral causality containing I/C elements in derivative causality, the length of the causal path is: $l = \text{number of } I/C \text{ elements in integral causality} - \text{number of } I/C \text{ elements in derivative causality} + 1$ if the end of the path is an I/C element in integral causality.

The input/output path length presented in Figure 3.5 is equal to 1. Thus, the bicausality is propagated from $S_fS_e : \omega_m$ to $D_fD_e : \hat{W}_a$ (Figure 3.6). We remark that the causal path length from $MSf : W_b$ to $Df : i_m$ is also 1. Therefore, the bicausality could also be propagated from $S_fS_e : i_m$ to $D_fD_e : \hat{W}_b$

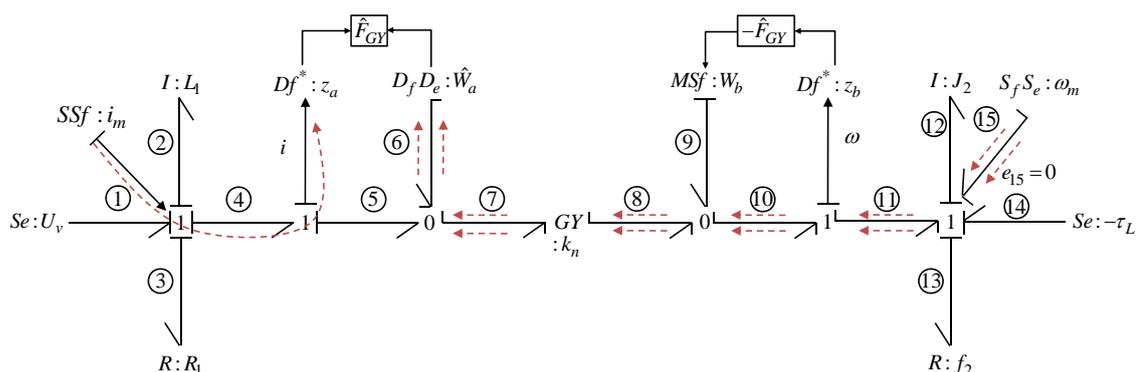


Figure 3.6: BG model of the DC-motor in bicausality for estimation of F_{GY} .

Remark. The bicausality could not be propagated from $S_fS_e : \omega_m$ to $SSf : \hat{W}_b$ because the rules of the bicausality affectation would be violated.

Local adaptive fault compensation

The fault estimation equations (3.7) are then obtained by covering the bicausal path as follows:

$$\begin{aligned}
 e_{11} &= e_{12} + e_{13} - e_{14} - e_{15}, & e_{15} &= 0, & e_{11} &= e_{10} = e_9 = e_8, \\
 e_8 &= J_2 \cdot \frac{d\omega_m}{dt} + f_2 \cdot \omega_m + \tau_L, & f_7 &= \frac{e_8}{k_n}, \\
 f_6 &= f_7 - f_5, & f_6 &= \frac{J_2 \cdot \frac{d\omega_m}{dt} + f_2 \cdot \omega_m + \tau_L}{k_n} - i_m, \\
 \hat{F}_{GY} &= \frac{f_6}{i_m},
 \end{aligned} \tag{3.7}$$

This reasoning enables to compensate not only the isolable faults but also non-isolable faults if the following structural conditions are respected.

- Let us define $D_{set} = \{Df_h, De_g | h + g = 1, \dots, v\}$, where v is the total number of detectors in the system. Therefore detectors can not belong to the triggered fault signature F_{sig}^b , which means the following: $D_{set} \cap F_{sig}^b = \emptyset$.
- When the bicausality is assigned for fault estimation, all storage elements belonging to the causal paths used for fault estimation must be in derivative causality.
- The faulty element is isolable or the junction is isolable. This means that we do not know the exact faulty location but all possible faults are associated to the same junction.

Hypothesis 3.2. The structure of the system does not change. This means that the faulty elements do not have to be removed from the BG model of the system.

The given conditions enable to extend the structural recoverability results presented in Chapter 2. Actually, the set of faults that can be compensated may increase.

Let us consider that the fault is not isolable and $D_{set} \cap F_{sig}^b = \emptyset$. Hence, to be able to compensate the fault with this technique, the triggered fault signature F_{sig}^b can only contain a set of the following elements ($R, I, C, Se, Sf, MSe, MSf$).

To obtain the additional power, the fault is modeled in one of the elements belonging to the triggered fault signature $\{R, I, C, Se, Sf, MSe, MSf\} \in F_{sig}^b$. To select it, we must take into account that the storage elements (I, C) do not affect the system during steady state periods. For this reason, the BG-LFT must not be applied to them if F_{sig}^b contains R or sources elements. For example, consider that a fault is presented in an R -element and that the BG-LFT for fault modeling is applied on an I element with the same fault signature. Since during steady state the estimated fault on the I -element is equal to zero $\hat{F}_I = 0$, the power generated by the fault could not be obtained during steady state operation. On the other hand, if the fault is modeled on an R -element when the actual fault is in the I -element, F_R would be equal to zero during the steady state ($F_R = 0$) and different from zero during transient periods $F_R \neq 0$. Thus, emulating correctly the fault on I .

Finally, for the type of non-isolable faults with isolable junction, the estimated power caused by the fault is injected onto this junction. To present the validity of the structural analysis results, we propose to use inverse model for control. This is used because it can be easily obtained from the BG model, and it is able to consider the fault effects in a straight forward manner.

3.3 Model inversion

Contrarily to the direct model, an inversion of the direct model is used to know which inputs are required given the desired outputs. The BG modeling tool has proven to be a powerful tool not only for modeling the direct model of a system but also to obtain its inverse model. In fact, the BG model inversion is done by the bicausal notion. The idea of using the bicausality for system inversion is motivated by its computation capabilities. As aforementioned, the bicausality enables to impose both power variables as an output of the $S_e S_f$ elements. A general representation of the BG model for direct and inverse system is presented in Figure 3.7 and 3.8, respectively.

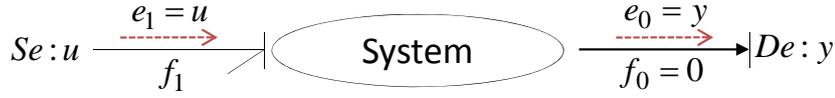


Figure 3.7: General representation of a BG direct model.

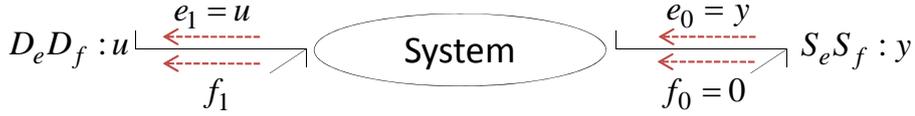


Figure 3.8: General representation of a BG inverse model.

In the direct BG model depicted in Figure 3.7, the system receives $e_1 = u$ as an input, while the measured output is $e_0 = y$. Hence, the power variable of flow (f_1) conjugated with the power variable of effort (e_1), is deduced from the system. On the other hand, if we desire to obtain the system inversion (Figure 3.8), the power variable $e_0 = y$ and its conjugated f_0 are the inputs of the inverse system. However, in order to do not furnish power to the system, the input flow must be set to zero ($f_0 = 0$). After propagation of the bicausality from output to input, we reach the input variable $e_1 = u$ while leaving f_1 free, being an output of the inverse system.

The procedure 3.1 presented in [Fotsu-Ngwompo 1997, Jardin 2010], enables to propagate the bicausality in order to construct the inverse model of a system with m inputs and m outputs. Before recalling the procedure of the system inversion, let us introduce some helpful definitions:

Definition 3.2. [Wu 1995] In an acausal bond graph model, a power line between two components is a series of power bonds and junctions structure elements connecting these two components.

Definition 3.3. An input output (I/O) power line is a power line between two BG elements, in which one element contain an input variable and the other element contain an output one.

Definition 3.4. ([Fotsu-Ngwompo 1997]) Two I/O causal paths are said to be disjoint

if they have no common variable.

Invertibility conditions can be found in [Jardin 2008] and it consists in finding:

1. At least one set of input/output disjoint power lines;
2. and, at least on set of input/output disjoint causal paths,

if the requires sets exist, then it can be concluded that the model is invertible.

Procedure 3.1. *Bicausality assignment for model inversion (procedure SCAPI)*
 ([Fotsu-Ngwompo 1997])

1. *In the BG model with preferred integral causality, select a set of disjoint I/O power lines associated with a set of disjoint I/O causal paths. If such sets do not exist, the model is not invertible and the procedure stops.*
2. *In an acausal BG model, replace the source elements (and detectors) associated to the inputs (and outputs) by double detector DeDf (and double source SeSf).*
3. *For each element of which causality is imposed (sources, elements with non-invertible constitutive laws) assign it with its causality and propagate the causality through the junction structure, taking into account the causality constraints of 0- and 1-junctions and GY-, TF- elements.*
4. *Along each power selected on the first step, propagate the bicausality from the double source to the double detector and propagate the causality through the junction structure, taking into account the causality constraints of 0- and 1-junctions and GY-, TF- elements.*
5. *Choose any energy storage element (C or I) without causality, and assign a preferred integral causality to it. Propagate the causality as far as possible. Re-do this step until all storage elements are causalised.*
6. *If some R-elements remain not causally determined, assign a causality to one and propagate it as previously. Repeat this step until all R-elements are causally determined.*

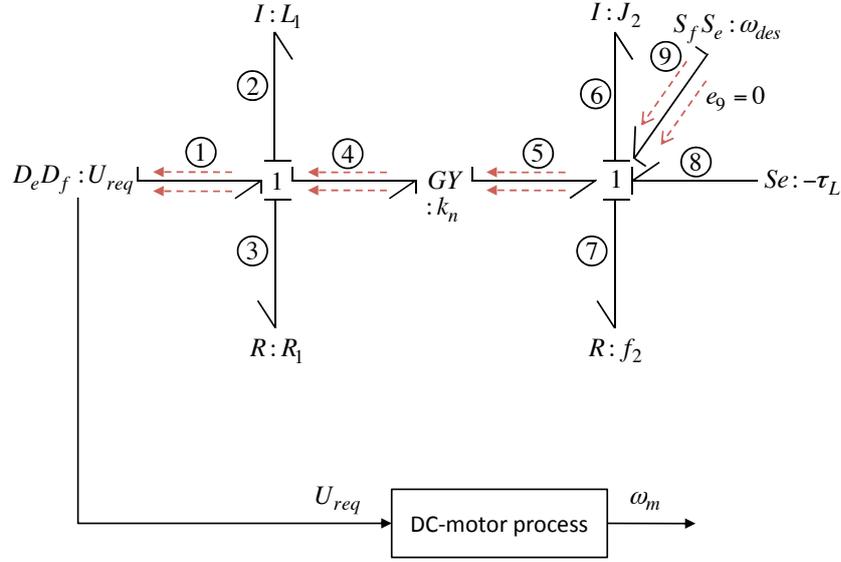


Figure 3.10: Inverse BG model of the DC-motor with the representation in the form of block diagram of the DC-motor physical process.

3.4 Bond graph fault tolerant control through power variable compensation

The concept of system inversion together with the supply of the power caused by the fault into BG model gives the basis to compute appropriate control actions that compensate the faults. The effects of the fault can then be compensated, by feeding the power generated by the fault in the correct place, so that the inverse system control is able to consider them. Logically, the ability to compensate the fault is dependent on the capacity of the actuators.

Remark. The introduction of the BG-LFT for fault modeling does not affect the invertibility of the system.

In addition, let us consider again the presence of a fault in the GY element. The inverse model is obtained by propagating the bicausality from $S_f S_e : \omega_{des}$ to $D_e D_f : U_{req}$, as depicted in Figure 3.11.

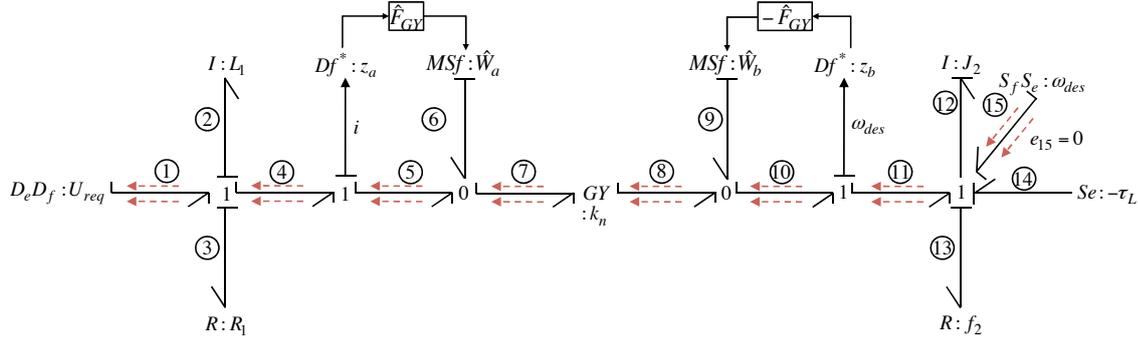


Figure 3.11: BG inverse model for GY fault compensation.

By covering the bicausal path, one can derive the required input (U_{req}) that furnishes the desired output (ω_{des}) as follows (3.8):

$$\begin{aligned}
 \omega_{des} &= f_{14} = f_{13} = f_{12} = f_{11} = f_{10}, & e_{11} &= e_{12} + e_{13} - e_{14} - e_{15}, & e_{15} &= 0, \\
 e_8 &= e_9 = e_{10} = e_{11}, & e_8 &= J_2 \frac{d\omega_{des}}{dt} + f_2 \cdot \omega_{des} + \tau_L \\
 f_9 &= \hat{W}_b = -\hat{F}_{GY} \cdot \omega_{des}, & f_8 &= f_{10} - f_9, \\
 e_7 &= f_8 \cdot k_n & f_7 &= \frac{e_8}{k_n}, & f_6 &= \hat{W}_a = \hat{F}_{GY} \cdot i, \\
 f_5 &= f_7 - f_6, & e_5 &= e_6 = e_7, & f_1 &= f_2 = f_3 = f_4 = f_5, \\
 U_{req} &= e_1 = e_2 + e_3 + e_4, & U_{req} &= L_1 \frac{df_2}{dt} + R_1 f_3 + e_4.
 \end{aligned} \tag{3.8}$$

Finally, when the bicausality is assigned, we must verify if the causality of the passive elements is not inverted, else the BG-LFT must be modified for system inversion. For instance, consider the R -element is conductive causality as depicted in Figure 3.12-(a) with its associated BG-LFT for fault modeling 3.12-(b).

If a bicausality is propagated through the junction associated to this R -element, the conductive causality changes automatically to a resistive one. This occurs to avoid a causal conflict (Figure 3.13-(a)). Therefore, the faulty modeling is also modified as presented in Figure 3.13-(b).

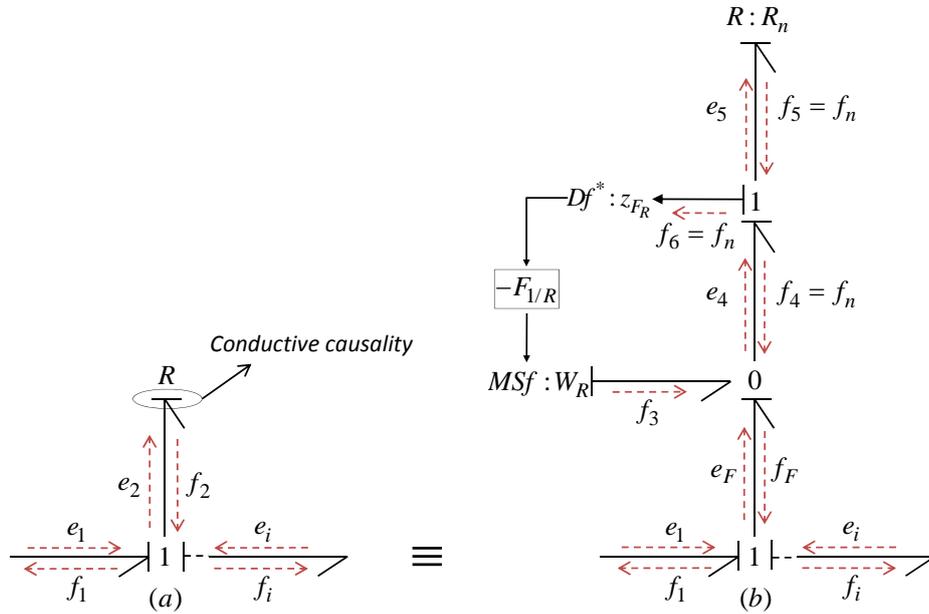


Figure 3.12: (a) - R -element in conductive causality associated to a 1-junction, (b) - BF-LFT for fault modeling of the R -element in conductive causality.

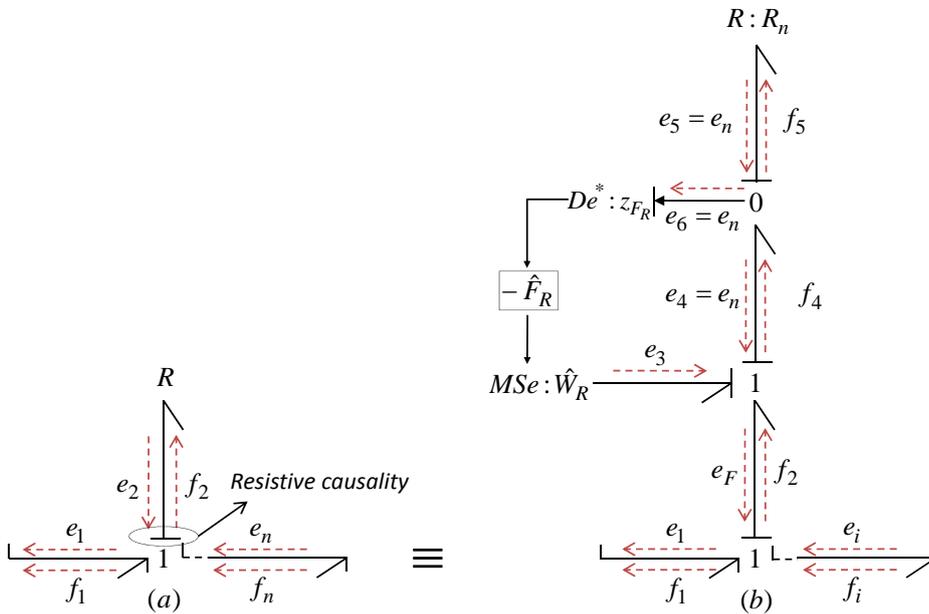


Figure 3.13: (a) - R -element in resistive causality associated to a bicaused 1-junction, (b) - BF-LFT for fault modeling of the R -element in resistive causality.

Bellow, it is presented the procedure for obtaining the inverse BG model of the system together with the fault effects.

Procedure 3.2. (*Procedure to obtain the inverse system with the fault effects*)

1. *From the acausal bond graph model obtain the inverse model,*
2. *Based on the fault information apply the BG-LFT as follows:*
 - *Fault is isolable. Apply the BG-LFT on the faulty parameter.*
 - *Junction is isolable and no sensor belongs to the triggered fault signature. Apply the BG-LFT on an R-element or a source associated to the "faulty" junction.*

For clarification purposes, we propose to perform simple simulation on a DC-motor with a fault in the inertial element ($I : L_1$) of the electrical part of the system. This fault is not isolable, as depicted in FSM (Table 2.4), the fault signature is as follows: $F_{sig} = [1, 0, 0, 0]$, hence $\{Se : U_v, R_1, L_1\} \in F_{sig}^b$. The fault is introduced at 20sec, and as depicted in Figure 3.15 when no compensation is performed, the system can not follow the desired angular velocity.

The junction is isolable, and the fault is modeled on the R_1 element using the procedure presented in Figure 3.13. The overall scheme of the proposed procedure in the form of block diagram is given in Figure 3.14. In this case, the required inputs (U_{req}) obtained from the inverse system are able to compensate the fault as depicted in Figure 3.16. The first graph depicts the tracked velocity, the second presents the tracking error while the third shows the estimation of the power generated by the power. As it can be concluded, even though the estimation of the power generated by the power is not performed in the exact faulty element, the fault effects are compensated.

3.5 Conclusions

In this chapter, we proposed a novel way to consider the fault recoverability with respect to fault compensation. By exploiting this idea, **we proposed a set of structural conditions that enable to verify the set of faults that can be compensated**

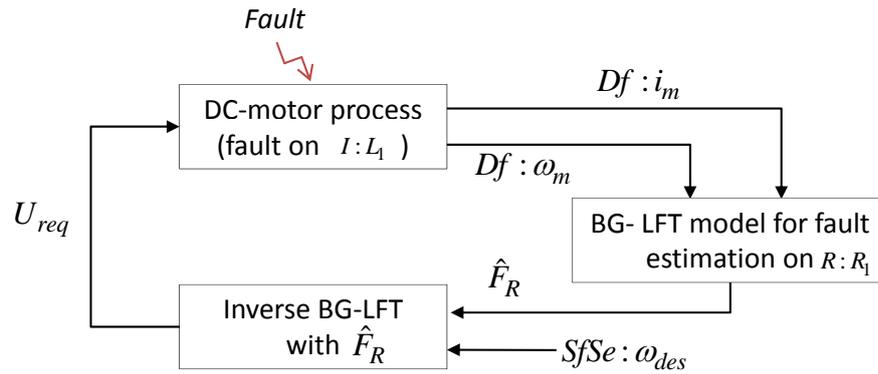


Figure 3.14: Local adaptive compensation scheme in the form of block diagram.

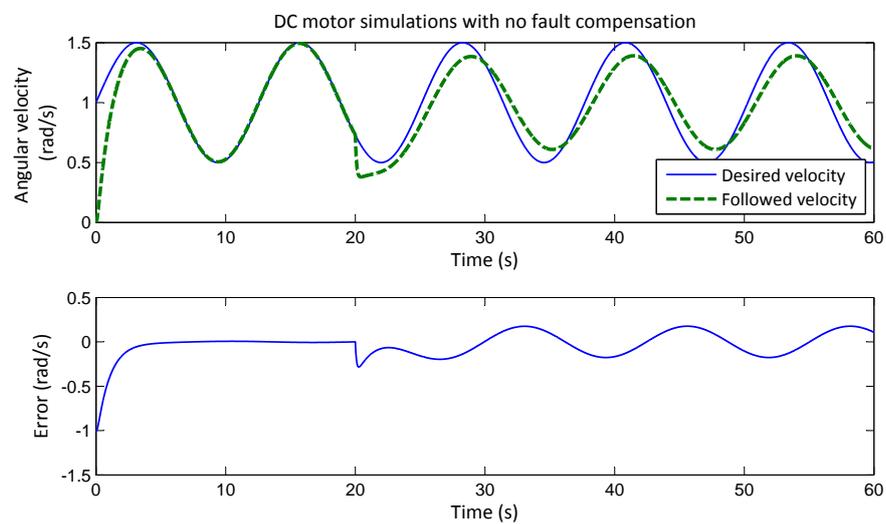


Figure 3.15: Simulation results when the fault is not compensated.

with this approach. To this end, the energetic structure of the BG graph is used to represent the faults as modulated sources that are known from the fault estimation procedure. This property of the bond graphs is interesting in the sense that as long as we know the power generated by the fault, component fault isolation is not a necessary condition. To validate the given structural results, a local adaptive fault compensation strategy is presented. Hence, once a bond graph model of the faulty system is obtained, the notion of bicausality is applied to obtain the inverse system. This inverse system is enabled to compute appropriate control actions that compensate the faults. Finally, further results of this method will be given in Chapter 4.

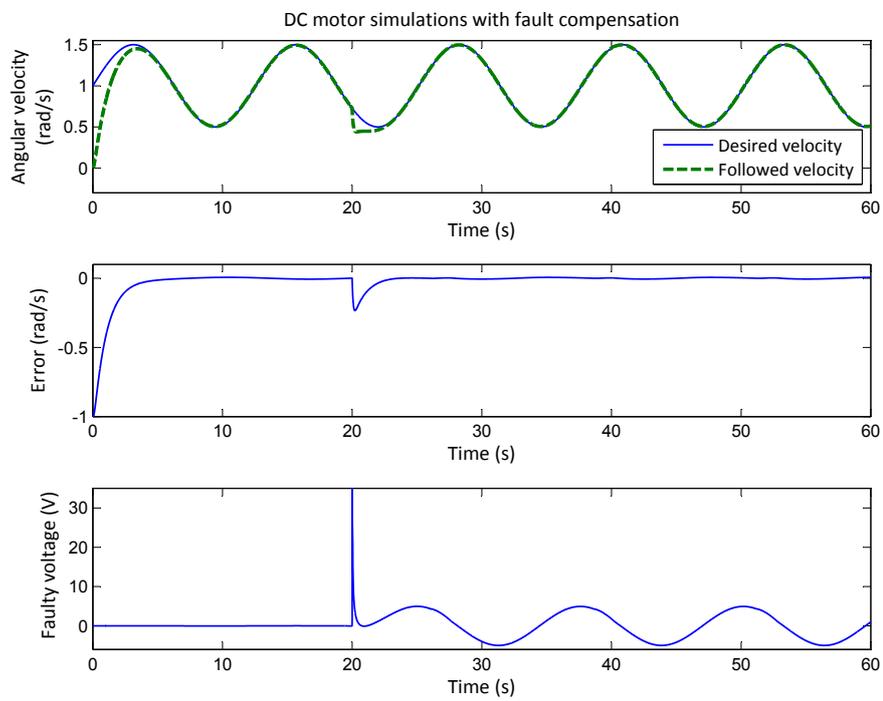


Figure 3.16: Simulation results when the fault is compensated.

Chapter 4

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

4.1 Introduction

This chapter is reserved to the validation of the contributions previously presented in Chapters 2 and 3 on a Multi-Input Multi-Output (MIMO) redundant Heavy-Sized Intelligent Autonomous vehicle (IAV), named RobuTAINeR. This vehicle has been conceived for the European project *InTraDE* (Intelligent Transportation for Dynamic Environment [InTraDE 2012]), and it is in its final development stage. Intelligent Transport Systems (ITS) are advanced applications that associate information and communication technologies (ICT) to vehicles and their infrastructures in order to increase safety, reliability, efficiency and reduce traffic. One of the main components belonging to ITS are intelligent autonomous vehicles (IAV), which can be used for the transportation of people or goods. The major objective of developing IAVs is to enable its safe operation inside confined and private areas, such as the seaport terminal depicted in Figure 4.1, or on existing roadway network without any human interaction. Hence, the road infrastructures do not have to be especially adapted to these vehicles. If the IAV are used in daily life, several advantages with respect to economical, environmental and society contexts can be presented. For instance, accidents, which are mostly caused by human errors, can be prevented since the controller algorithms are more reactive than human drivers. Traffic congestion can be improved, when the number of vehicles is dense according to road length because they can adapt their velocities and their trajectories according to the traffic and the environment status.

4.2 Intelligent autonomous vehicle description and modeling

RobuTAINeRs is an IAV designed to handle 20 ft containers and two identical vehicles can be coupled together in order to transport a 40 ft container. This IAV can carry either a container as directly deposited by an external handling system such as a crane or a container placed on a table with the table being taken as well as the container.

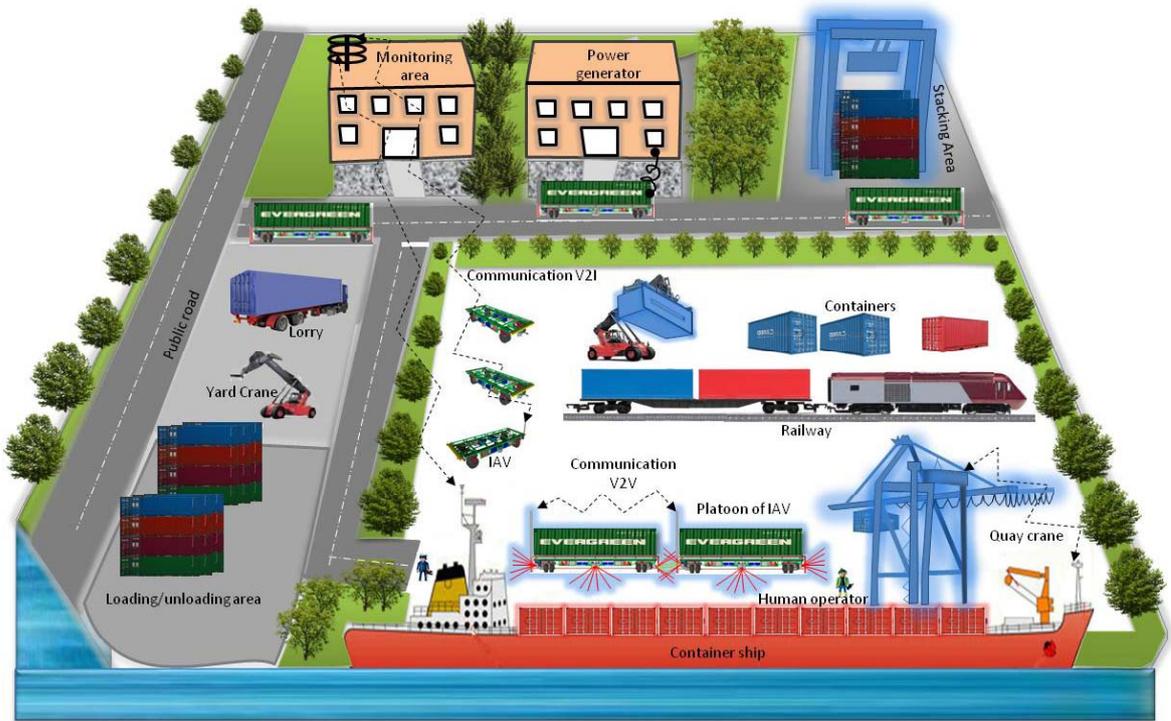


Figure 4.1: Confined space of port terminal.

The Figure of the IAV is depicted in Figure 4.2. It is composed of six main sub-systems, which are: vehicle body, air suspension, traction and braking wheel, steering motor, traction motor, and braking system. The rear and front wheels can be independently steered by four steering motors. Each driving wheel is driven independently and brakes can also be independently applied to the braking wheels.



Figure 4.2: (a) CAD figure of RobuTAINeR's prototype, (b) Real RobuTAINeR picture.

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

Each j^{th} wheel frame $j \in [1, 4]$ is coupled with both j^{th} traction and j^{th} braking wheels, as shown in Figure 4.3, which are then connected to the vehicle body by the air suspension. $j = 1$ and $j = 2$ are associated with the front left (FL) and front right (FR) subsystems, respectively. While $j = 3$ and $j = 4$ are associated with the rear left (RL) and rear right (RR) subsystems, respectively. The vehicle is controlled with four decentralized traction DC-motors, and steered with four independent steering electro-hydraulic motors. Moreover, the brakes are also independently applied on the dual wheels.

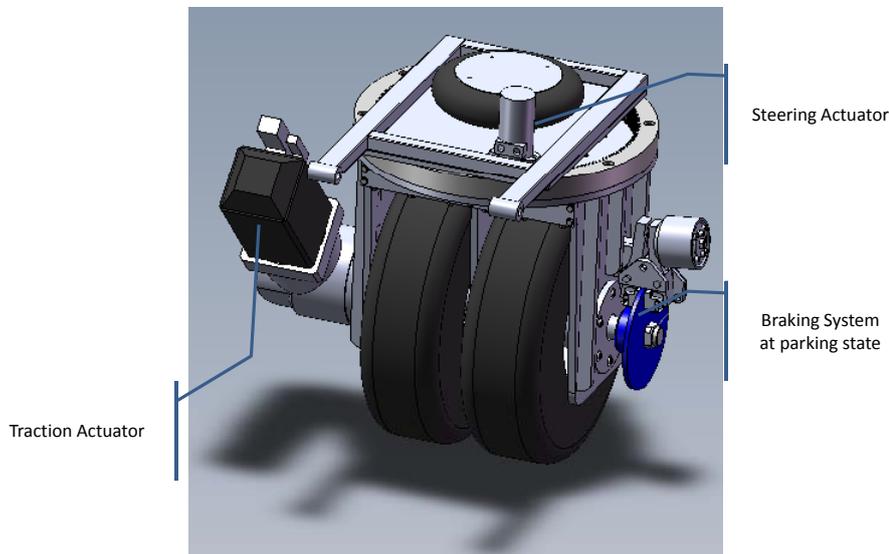


Figure 4.3: CAD figure of RobuTAINeRs wheel frame with steering and traction motor.

4.2.1 RobuTAINeRs quarter vehicle modeling

The considered dynamics in this work are the electromechanical systems for traction, together with the longitudinal, lateral and yaw dynamics of the chassis. The maximal velocity applied on the vehicle is $25km/h$, and the road surface is assumed uniform. Therefore, some dynamics have neglected effects on the whole vehicle motion, such as:

- Pitch and roll dynamics.
- Suspension dynamics.

Finally, the considered dynamics for this case study are the electromechanical model of the traction system, the longitudinal, lateral, and yaw dynamics on the center of gravity (CoG) of the vehicle.

The electromechanical traction subsystem is composed of four main parts as illustrated in the word bond graph in Figure 4.4, where the major components of the subsystem and their interconnections are represented. This components are a DC-motor, which is the combination of electrical and mechanical dynamics, a gear, and the wheel in interaction with the ground as depicted in Figure 4.5.



Figure 4.4: Word bond graph of a quarter RobuTAINeRs (electromechanical subsystem).

1. *Electrical part of the DC-motor:* It corresponds to an RL circuit composed of an input voltage U_{0j} , an electrical resistance R_{ej} , and an inductance L_{ej} . In the BG model, these components are modeled, respectively with a Se , R , and I elements. Moreover, there is also an electromotive force feedback EMF (with a constant k_{ej}) that is modeled with a GY element.
2. *Mechanical part of the DC-motor:* It is characterized by the rotor inertia J_{ej} , a viscous friction parameter f_{ej} , and the shaft rigidity K_j . In the BG model, these components are modeled, respectively with a I , R , and C elements.
3. *Gear part:* It is defined by reduction constant N_j , and it links the mechanical part of the DC-motor to the wheel. This part is represented in a BG model by a TF element.
4. *Wheel + ground part:* The wheel in interaction with the ground through the tire represents the load of the electromechanical system. It is characterized by its inertia $I : J_{sj}$, a viscous friction parameter $R : f_{sj}$ the longitudinal contact effort

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

$MSe : Fl_j$, which is a modulated and a nonlinear identified effort source. Finally, the radius is defined by r and it is considered static and the same for all the wheels.

The measurement architecture of these subsystems is composed of a current (i_j), the angular position of the rotor (θ_{ej}), and the angular position of the wheel (θ_{sj}) sensors. They are represented on the BG model by i_{mj} , and the angular velocities ($\dot{\theta}_{ej}$, and $\dot{\theta}_{sj}$) computed from the measurement of the positions.

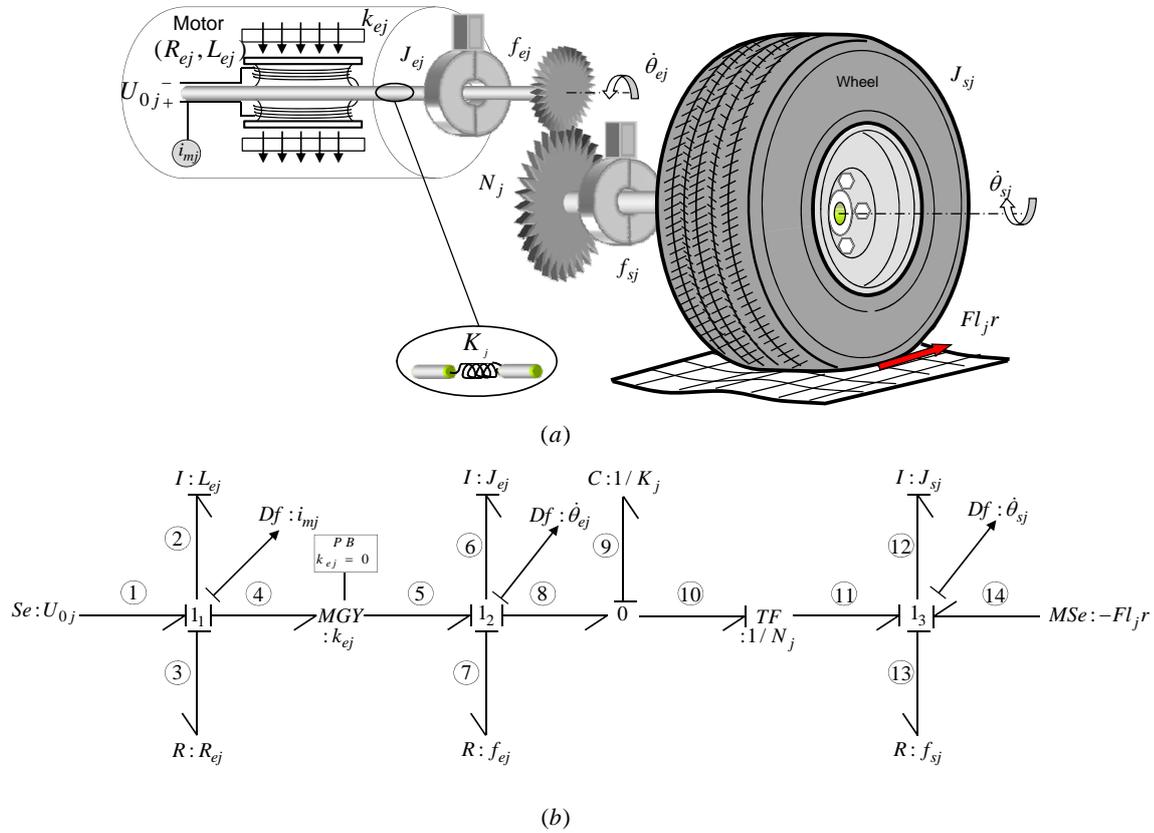


Figure 4.5: Bond graph of a quarter RobuTAINeRs (electromechanical subsystem).

The state space equations of the j^{th} electromechanical subsystem can be deduced from the BG model. The number of state variables is equal to the number of dynamic elements I and C in integral causality. From Figure 4.5, one can conclude the existence of four state variables. The state equations of the j^{th} electromechanical system appear below as equation (4.16), where p_{Lj} , p_{Jj} , q_{Kj} and p_{Sj} are the energy variables of L_{ej} , J_{ej} , K_j , and J_{sj} elements, respectively.

The development of the dynamic equations from the BG model of Figure 4.5 is presented below.

Constraints of structure C_S

1_1 - junction :

$$\begin{aligned} f_1 = f_2 = f_3 = f_4 = i_j, \\ e_1 - e_2 - e_3 + e_4 = 0. \end{aligned} \tag{4.1}$$

1_2 - junction :

$$\begin{aligned} f_5 = f_6 = f_7 = f_8 = \dot{\theta}_{ej}, \\ e_5 - e_6 - e_7 - e_8 = 0. \end{aligned} \tag{4.2}$$

0 - junction :

$$\begin{aligned} e_8 = e_9 = e_{10}, \\ f_8 - f_9 - f_{10} = 0. \end{aligned} \tag{4.3}$$

1_3 - junction :

$$\begin{aligned} f_{11} = f_{12} = f_{13} = f_{14} = \dot{\theta}_{sj}, \\ e_{11} - e_{12} - e_{13} + e_{14} = 0. \end{aligned} \tag{4.4}$$

Gyrator $GY : k_{ej}$

$$\begin{aligned} e_5 = f_4 k_{ej}, \\ e_4 = f_5 k_{ej}. \end{aligned} \tag{4.5}$$

Transformer $TF : 1/N_j$

$$\begin{aligned} f_{10} = f_{11} N_j, \\ e_{11} = e_{10} N_j. \end{aligned} \tag{4.6}$$

Behavioral equations C_B

$$I : L_{ej} - Element : f_2 = \frac{1}{L_{ej}} \int e_2, \quad (4.7)$$

$$I : J_{ej} - Element : f_6 = \frac{1}{J_{ej}} \int e_6, \quad (4.8)$$

$$C : K_j - Element : e_9 = K_j \int f_9, \quad (4.9)$$

$$I : J_{sj} - Element : f_{12} = \frac{1}{J_{sj}} \int e_{12}. \quad (4.10)$$

$$R : R_{ej} - Element : e_3 = R_{ej} f_3, \quad (4.11)$$

$$R : f_{ej} - Element : e_7 = f_{ej} f_7, \quad (4.12)$$

$$R : f_{sj} - Element : e_{13} = f_{sj} f_{13}, \quad (4.13)$$

Knowing that:

$$\begin{aligned} x_1 = p_{Lj} &= \int e_2, & x_2 = p_{Jj} &= \int e_6, \\ x_3 = q_{Kj} &= \int f_9, & x_4 = p_{Sj} &= \int e_{12}. \end{aligned} \quad (4.14)$$

The following dynamic equations are obtained:

$$\begin{aligned} \dot{p}_{Lj} = e_2 &= -\frac{R_{ej}}{L_{ej}} p_{Lj} - \frac{k_{ej}}{J_{ej}} p_{Jj} + U_{0j}, \\ \dot{p}_{Jj} = e_6 &= \frac{k_{ej}}{L_{ej}} p_{Lj} - \frac{f_{ej}}{J_{ej}} p_{Jj} - K_j q_{Kj}, \\ \dot{q}_{Kj} = f_9 &= \frac{1}{J_{ej}} p_{Jj} - \frac{N_j}{J_{sj}} p_{Sj}, \\ \dot{p}_{Sj} = e_{12} &= NK_j q_{Kj} - \frac{f_{sj}}{J_{sj}} p_{Sj} - Fl_{j.r}. \end{aligned} \quad (4.15)$$

This can be represented in the state space format as follows (4.16) :

$$\begin{aligned}
 \underbrace{\begin{bmatrix} \dot{p}_{Lj} \\ \dot{p}_{Jj} \\ \dot{q}_{Kj} \\ \dot{p}_{Sj} \end{bmatrix}}_X &= \underbrace{\begin{bmatrix} -\frac{R_{ej}}{L_{ej}} & -\frac{k_{ej}}{J_{ej}} & 0 & 0 \\ \frac{k_{ej}}{L_{ej}} & -\frac{f_{ej}}{J_{ej}} & -K_j & 0 \\ 0 & \frac{1}{J_{ej}} & 0 & -\frac{N_j}{J_{sj}} \\ 0 & 0 & N_j K_j & -\frac{f_{sj}}{J_{sj}} \end{bmatrix}}_A \underbrace{\begin{bmatrix} p_{Lj} \\ p_{Jj} \\ q_{Kj} \\ p_{Sj} \end{bmatrix}}_X + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix}}_B \underbrace{\begin{bmatrix} U_{0j} \\ Fl_j \cdot r \end{bmatrix}}_U, \\
 \underbrace{\begin{bmatrix} i_{mj} \\ \dot{\theta}_{ej} \\ \dot{\theta}_{sj} \end{bmatrix}}_Y &= \underbrace{\begin{bmatrix} \frac{1}{L_{ej}} & 0 & 0 & 0 \\ 0 & \frac{1}{J_{ej}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{J_{sj}} \end{bmatrix}}_C \underbrace{\begin{bmatrix} p_{Lj} \\ p_{Jj} \\ q_{Kj} \\ p_{Sj} \end{bmatrix}}_X.
 \end{aligned} \tag{4.16}$$

After generalization of the quarter dynamic model for traction, the overall BG model of the vehicle with the considered dynamics is illustrated in Figure 4.6. In addition to the traction dynamic models, this model also considers the longitudinal, lateral and yaw dynamics. Where, a and b are respectively the distances from the vehicle CoG to the front and the rear axles, and c is the distance of the left and right wheels from the longitudinal vehicle axis.

F_{c_j} is the cornering force transmitted to the wheel, and it is calculated in [Soc 2001] and given as follows.

$$F_{c_j} = \frac{m(\dot{\theta}_{sj} \cdot r)^2}{d}, \tag{4.17}$$

where d is the radius of the bend, and m the mass of the vehicle. Moreover, Fl_j is the longitudinal contact effort and is generated as a function of the longitudinal velocity (\dot{u}), and of the angular one ($\dot{\theta}_{sj}$), as illustrated by the canonical curve estimated from the nonlinear model of [Pacejka 1991], and represented by (4.19).

$$Fl_j = [\delta_0 - \delta_1 e^{-\beta(|\dot{u} - r\dot{\theta}_{sj}|)} - \delta_2(\dot{u} - r\dot{\theta}_{sj})] \cdot \text{sign}(\dot{u} - r\dot{\theta}_{sj}), \tag{4.18}$$

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

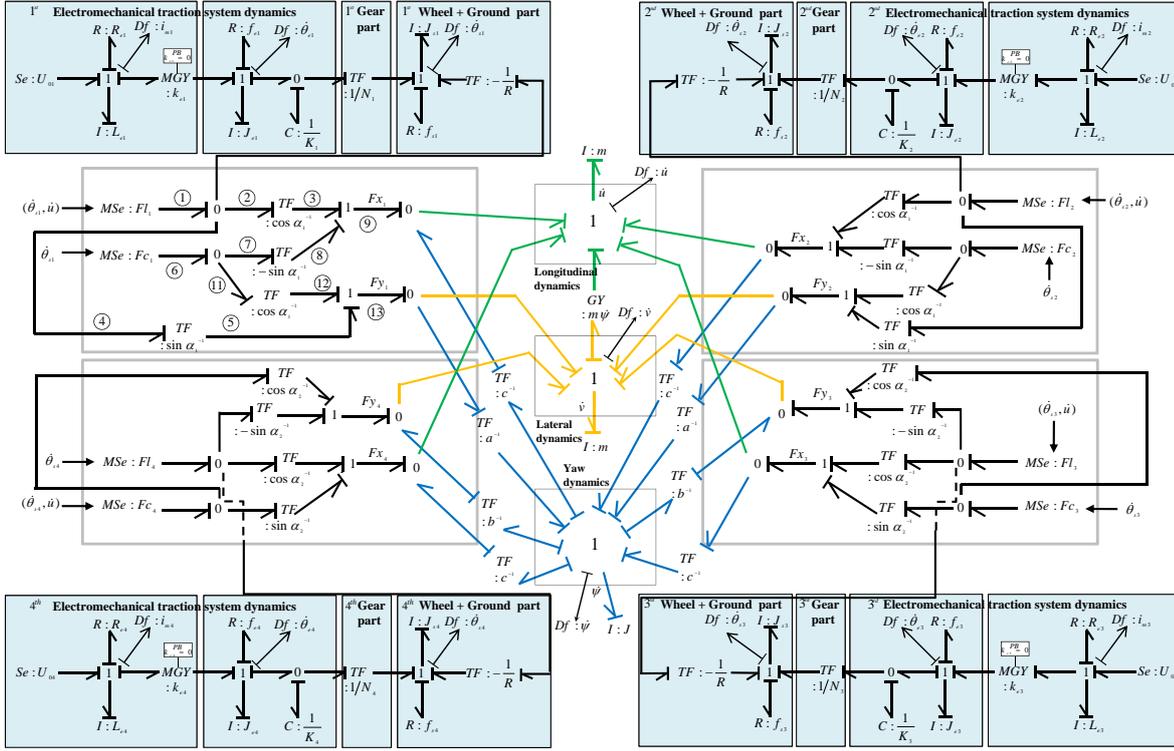


Figure 4.6: BG model of RobuTAINER's.

where δ_0 , and δ_1 are the dry and the stiction forces [N], respectively. δ_2 is the viscous coefficient [N°/s]. β is the stiction coefficient. Knowing that the slip velocity ($\dot{u} - R\dot{\theta}_{sj}$) is small at a maximum operated velocity of CoG (25Km/h), then the stictius and viscous behaviors of Fl_j are neglected. Thus equation (4.19) is simplified to coulomb friction behavior [R. Merzouki 2007], as follows:

$$Fl_j = \delta_0 \cdot \text{sign}(\dot{u} - r\dot{\theta}_{sj}). \quad (4.19)$$

Fx_1 , and Fy_1 are computed by following the paths from bonds ① and ②, to bonds ⑨ and ⑬ of Figure 4.6.

$$Fx_1 = Fl_1 \cos(\alpha_1) - Fc_1 \sin(\alpha_1),$$

$$Fy_1 = Fl_1 \sin(\alpha_1) + Fc_1 \cos(\alpha_1).$$

The same can be obtained for the remaining forces. From the BG model one can also obtain the mathematical expressions that model the vehicle motion of the longitudinal dynamics (4.20), the lateral one is presented in (4.21), and the yaw in (4.22). The terms $m.\dot{\psi}.\dot{v}$ and $m.\dot{\psi}.\dot{u}$ are respectively the effects of the yaw dynamics in the longitudinal and lateral directions.

$$m.\ddot{u} = Fx_1 + Fx_2 + Fx_3 + Fx_4 + m.\dot{\psi}.\dot{v}, \quad (4.20)$$

$$m.\ddot{v} = Fy_1 + Fy_2 + Fy_3 + Fy_4 - m.\dot{\psi}.\dot{u}, \quad (4.21)$$

$$J.\ddot{\psi} = [-Fx_1 + Fx_2 + Fx_3 - Fx_4].c + [Fy_1 + Fy_2].a - [Fy_3 + Fy_4].b. \quad (4.22)$$

4.3 Structural analysis for fault tolerance

Structural analysis of fault tolerance require the following steps:

- Associate the system objectives to the BG model;
- Perform fault diagnosis to obtain the FSM, and locate the existence of PBs;
- From the different fault signatures evaluate the level of fault tolerance.

The desired objective (Σ_o) are defined as driving the RobuTAINeR at a desired longitudinal, lateral, and yaw velocities ($\dot{u}^d, \dot{v}^d, \dot{\psi}^d$). The diagnosis procedure is presented in the following subsection.

4.3.1 Diagnosis procedure

Recalling that to generate *ARRs* from the BG model, the system must be in derivative causality and the detectors dualized if possible. This procedure is applied to the BG model of the electromechanical subsystem (Figure 4.5) and the following model depicted in Figure 4.7 is obtained.

We remark that the element used to model the shaft rigidity ($C : \frac{1}{K_j}$) do not accept a derivative causality. However, since its initial conditions are known ($\theta_{ej}(0) = \theta_{ej}(0) = 0$), it can be presented in integral causality.

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

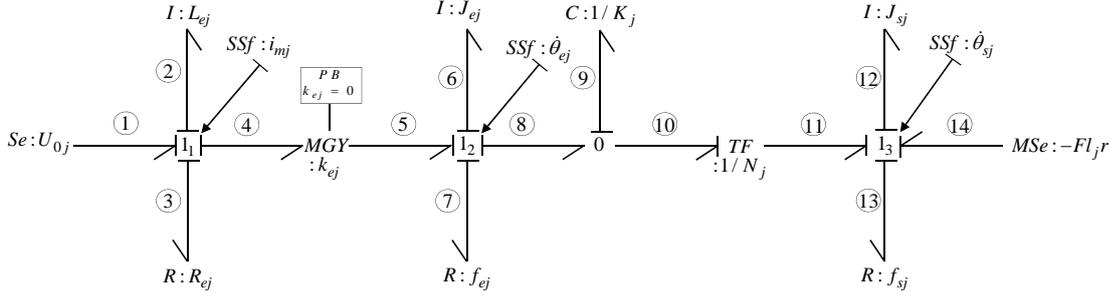


Figure 4.7: BG model of the j^{th} electromechanical subsystem in derivative causality.

By applying the causality inversion method, the following *ARRs* are obtained.

$$ARR_{1j}: U_{0j} - L_{ej} \frac{di_{mj}}{dt} - R_{ej} i_{mj} - k_{ej} \dot{\theta}_{emj} = 0, \quad (4.23)$$

$$ARR_{2j}: k_{ej} i_{mj} - f_{ej} \dot{\theta}_{ej} - J_{ej} \frac{d\dot{\theta}_{ej}}{dt} - K_j (\theta_{ej} - \theta_{sj} N_j) = 0, \quad (4.24)$$

$$ARR_{3j}: N_j K_j (\theta_{ej} - N_j \theta_{sj}) - J_{sj} \frac{d\dot{\theta}_{sj}}{dt} - f_{sj} \dot{\theta}_{sj} - Fl_{jr} = 0. \quad (4.25)$$

From the structure of these *ARRs*, the fault signature matrix (FSM) presented in Table 4.1 is deduced.

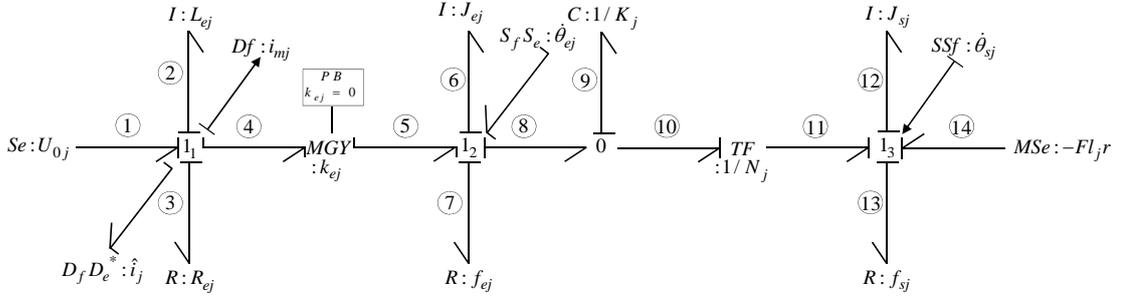
We remark that with these *ARRs*, the detector $\dot{\theta}_{ej}$ is the only fault that can be isolated. To improve these results we apply the procedure detailed in Chapter 2, Subsection 2.4.6.

From the first three columns of the FSM (Table 4.1), it is possible to conclude that the detector $Df : i_{mj}$ is not isolable. $ARR_{1j} : f_{1j}(i_{mj}, \dot{\theta}_{ej}, \dots) = 0 \wedge ARR_{2j} : f_{2j}(i_{mj}, \dot{\theta}_{ej}, \dot{\theta}_{sj}, \dots) = 0$. Then, it is possible to compute an extra *ARR* (ARR_{4j}), without using $Df : i_{mj}$ for diagnosis, so that $ARR_{4j} : f_{4j}(\dot{\theta}_{ej}, \dot{\theta}_{sj}, \dots) = 0$.

To this end, the virtual double detector $D_f D_e^* : \hat{i}$ is added to the junction associated with $Df : i_{mj}$, and $SSf : \dot{\theta}_{ej}$ is replaced by a double source $S_f S_e : \dot{\theta}_{ej}$. Finally, the bicausality is propagated from $S_f S_e : \dot{\theta}_{ej}$ to $D_f D_e^* : \hat{i}$, as depicted in Figure 4.8, in order to estimate the current (\hat{i}). This estimation is used to compute ARR_{4j} .

Table 4.1: Fault Signature Matrix (FSM) of the j^{th} electromechanical system.

Part	Comp.	Residuals			D_b	I_b
		r_{1j}	r_{2j}	r_{3j}		
Electrical	$Se : U_{0j}$	1	0	0	1	0
	$I : L_{ej}$	1	0	0	1	0
	$R : R_{ej}$	1	0	0	1	0
	$Df : i_{mj}$	1	1	0	1	0
	$GY : k_{ej}$	1	1	0	1	0
Mechanical	$Df : \dot{\theta}_{ej}$	1	1	1	1	1
	$R : f_{ej}$	0	1	0	1	0
	$I : J_{ej}$	0	1	0	1	0
	$C : K_j$	0	1	1	1	0
	$TF : N_j$	0	1	1	1	0
Load (Tyre+Ground)	$Df : \dot{\theta}_{sj}$	0	1	1	1	0
	$R : f_{sj}$	0	0	1	1	0
	$I : J_{sj}$	0	0	1	1	0
	$TF : r$	0	0	1	1	0
	$MSe : Fl_j$	0	0	1	1	0


Figure 4.8: BG model of the j^{th} electromechanical subsystem in bicausality for estimation of current \hat{i}_j .

$$\hat{i}_j = \left(J_{ej} \frac{d\dot{\theta}_{ej}}{dt} + f_{ej} \dot{\theta}_{ej} + K_j (\theta_{ej} - N\theta_{sj}) \right) \frac{1}{k_{ej}}. \quad (4.26)$$

Finally, in order to obtain ARR_{4j} (4.27), i_{mj} is replaced by \hat{i}_j in ARR_{1j} .

$$ARR_{4j}: U_{0j} - R_{ej} \hat{i}_j - L_{ej} \frac{d\hat{i}_j}{dt} - k_{ej} \dot{\theta}_{ej} = 0. \quad (4.27)$$

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

Moreover, from the first four columns of Table 4.2, the detector $Df : \dot{\theta}_{sj}$ is not isolable. $ARR_{2j} : f_{2j}(i_{mj}, \dot{\theta}_{ej}, \dot{\theta}_{sj}, \dots) = 0 \wedge ARR_{3j} : f_{3j}(\dot{\theta}_{ej}, \dot{\theta}_{sj}, \dots) = 0$. Then it is possible to compute an extra ARR (ARR_{5j}), from the junction associated to $SSf : \dot{\theta}_{sj}$, so that $ARR_{5j} : f_{5j}(\dot{\theta}_{im}, \dot{\theta}_{ej}, \dots) = 0$.

$$e_8 = k_{ej}i_m - \dot{\theta}_{ej}f_{ej} - J_{ej}\frac{d\dot{\theta}_{ej}}{dt}, \quad (4.28)$$

$$\hat{\theta}_{sj} = \frac{1}{N_j}(\dot{\theta}_{ej} - \frac{1}{K_j}\frac{de_8}{dt}), \quad (4.29)$$

$$ARR_{5j} : N_jK_j(\theta_{ej} - N_j\hat{\theta}_{sj}) - J_{sj}\frac{d\hat{\theta}_{sj}}{dt} - f_{sj}\hat{\theta}_{sj} - Fl_jr = 0. \quad (4.30)$$

These five $ARRs$ are used to compute a new FSM, illustrated in Table 4.2. We remark that all the system components are detectable and the set of isolable faults is the following: $[Df : i_{mj}, k_{ej}, Df : \dot{\theta}_{emj}, Df : \dot{\theta}_{smj}]$. Therefore, with these extensions, three new faults are isolable.

In addition, we remark the presence of a possible PB in each j^{th} electromechanical system. We consider that the converter of electrical energy to mechanical torque may provoke this situation, because (k_{ej}) is considered controlled. Then, it is possible to deduce that by setting the component $MGY : k_{ej}$ to zero ($k_{ej} = 0$), the dynamics of electrical subsystem are removed from the mechanical one of DC-motor. This can also be mathematically represented in (4.16). Actually, if $k_{ej} = 0$, the state p_L do not affect the ones of state p_J , and vice-versa. Physically, here the PB corresponds to the absence of the magnetic field that generates the mechanical torque.

Once this analysis have been performed, the evaluation of fault tolerance with respect to the different fault signatures is performed.

4.3.2 Structural fault tolerance analysis procedure

To detail the procedure of the algorithm of Figure 2.17, given in Chapter 2, two fault scenarios are considered:

Table 4.2: Fault Signature Matrix (FSM) of the j^{th} electromechanical system.

Part	Comp.	Residuals					D_b	I_b
		r_{1j}	r_{2j}	r_{3j}	r_{4j}	r_{5j}		
-	-						-	-
Electrical	$Se : U_{0j}$	1	0	0	1	0	1	0
	$I : L_{ej}$	1	0	0	1	0	1	0
	$R : R_{ej}$	1	0	0	1	0	1	0
	$Df : i_{mj}$	1	1	0	0	1	1	1
	$GY : k_{ej}$	1	1	0	1	1	1	1
Mechanical	$Df : \theta_{ej}$	1	1	1	1	1	1	1
	$R : f_{ej}$	0	1	0	1	1	1	0
	$I : J_{ej}$	0	1	0	1	1	1	0
	$C : K_j$	0	1	1	1	1	1	0
	$TF : N_j$	0	1	1	1	1	1	0
Load (Tyre+Ground)	$Df : \theta_{sj}$	0	1	1	1	0	1	1
	$R : f_{sj}$	0	0	1	0	1	1	0
	$I : J_{sj}$	0	0	1	0	1	1	0
	$TF : r$	0	0	1	0	1	1	0
	$MSe : Fl_j$	0	0	1	0	1	1	0

- **First scenario:** Fault on the input source of the front left electromechanical subsystem ($Se : U_{01}$).
- **Second scenario:** Plant fault on the viscous friction parameter of the wheel ($R : f_{s1}$).

The considered faults in these analyses are only contained in the electromechanical traction system of the RobuTAINeR. Nevertheless, the analysis of recoverability is provided based on its influences on the global vehicle. Two single fault scenarios were considered.

First scenario: Let us start by considering an input fault $Se : U_{01}$ introduced in the front left actuated subsystem. To verify the structural ability of the system to cope with this fault signature, the algorithm (Figure 2.17) begins and then it follows the following steps:

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

1. From step ①, the FDI information is obtained and the triggered fault signature is $F_{sig}^b = [1, 0, 0, 1, 0]$.
2. The elements $Se : U_{01}$, R_{e1} , and L_{e1} have the same triggered fault signature ($V_{U_{01}} = V_{R_{e1}} = V_{L_{e1}} = [1, 0, 0, 1, 0]$). Thus, the answer from step ② is "No" because we are not able to isolate either an actuator or a sensor fault.
3. Then the algorithm follows to ②② and, since the fault is not isolable, the answer is also "No".
4. The next step ②③, the algorithm searches a PB.
5. Having in mind PB conditions (1), and (2) presented in Chapter 2, a PB is found as illustrated in Figure 4.9. So the dynamics of the unknown faulty element can be removed from Σ_o . Consequently, the answer of step ②④ is "Yes".
6. In step ②⑤, Σ' is created.
7. Finally, in step ②⑥, the system must remain controllable, observable and monitorable. The healthy 1st electromechanical system does not have a directed actuated input. Nevertheless, the contact effort (Fl_1) is an undirected source. Moreover, $I : m$, and $I : J$ remain controllable with the remaining healthy actuated system, hence the answer of this step is "Yes."
8. It is then concluded from a structural point of view that the existing fault can be handled by applying a system reconfiguration technique.

Second scenario: In this case, let us consider a fault in the viscous friction parameter f_{sj} . Again, the algorithm is initialized, and it follows the following steps:

1. From step ①, the FDI information is obtained and the triggered fault signature is $F_{sig}^b = [0, 0, 1, 0, 1]$ and the set of elements belonging to it are: $F_{sig}^b = \{R : f_{sj}, I : J_{sj}, MSe : Fl_j, TF : r\}$.
2. The algorithm follows to step ② and its output is "No".

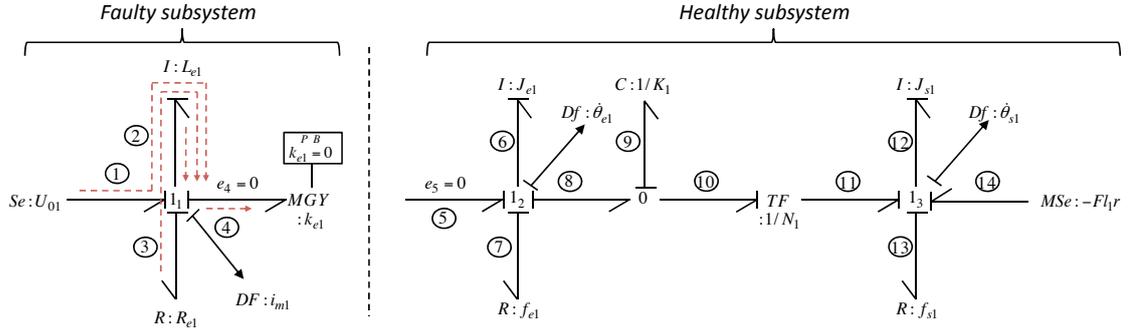


Figure 4.9: BG of the 1st electromechanical system with a PB label in the $MGY : k_{e1}$ element.

3. The next step is (22) and, since the fault is not isolable, the answer is also "No".
4. The search of a PB is done the step (23), and in this case, it is not found. Hence, the dynamics of the unknown faulty element can not be prevented from affecting the Σ_o .
5. Indeed, since the unknown fault dynamics keep affecting the system objectives, it is not possible to assure safety if the system is kept under operation.

By applying this procedure offline, to the complete set of possible fault signatures, the following Fault Signature and recoverability Matrix (FSRM) is obtained (Table 4.3). We notice that, the detector $Df : \theta_{sj}$ represents a control sensor, while the detectors $Df : i_{mj}$, $Df : \dot{\theta}_{ej}$, model diagnosis sensors. Therefore, a fault in either $Df : i_{mj}$, or $Df : \dot{\theta}_{ej}$ do not affect the ability of the system to be controlled. Hence, we introduce a (-) in its corresponding entry of SR and SA columns in the FSRM (Table 4.3).

From Table 4.3, one can define the set of critical fault signatures CF_{sig} , and the set of critical fault elements CF_{ele} , as follows:

$$\begin{cases} CF_{sig} = \{[0, 1, 0, 1, 1], [0, 1, 1, 1, 1], [0, 0, 1, 0, 1]\} \\ CF_{ele} = \{f_{ej}, J_{ej}, C : K_j, N_j, f_{sj}, J_{sj}, Fl_j, r\}. \end{cases} \quad (4.31)$$

We remark that fault accommodation can only be performed in the gyrator ($GY : k_{ej}$) element and on the wheel velocity sensor ($\dot{\theta}_{sj}$). This is because, in both of them,

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

Table 4.3: Fault Signature and recoverability Matrix (FSRM) of the j^{th} electromechanical system.

Part	Comp.	Residuals					D_b	I_b	SR	SA
		r_{1j}	r_{2j}	r_{3j}	r_{4j}	r_{5j}				
-	-						-	-	-	-
Electrical	$Se : U_{0j}$	1	0	0	1	0	1	0	1	0
	$I : L_{ej}$	1	0	0	1	0	1	0	1	0
	$R : R_{ej}$	1	0	0	1	0	1	0	1	0
	$Df : i_{mj}$	1	1	0	0	1	1	1	-	-
	$GY : k_{ej}$	1	1	0	1	1	1	1	1	1
Mechanical	$Df : \theta_{ej}$	1	1	1	1	1	1	1	-	-
	$R : f_{ej}$	0	1	0	1	1	1	0	0	0
	$I : J_{ej}$	0	1	0	1	1	1	0	0	0
	$C : K_j$	0	1	1	1	1	1	0	0	0
	$TF : N_j$	0	1	1	1	1	1	0	0	0
Load (Tyre+Ground)	$Df : \theta_{sj}$	0	1	1	1	0	1	1	0	1
	$R : f_{sj}$	0	0	1	0	1	1	0	0	0
	$I : J_{sj}$	0	0	1	0	1	1	0	0	0
	$TF : r$	0	0	1	0	1	1	0	0	0
	$MSe : Fl_j$	0	0	1	0	1	1	0	0	0

the fault can be isolated and estimated. However, estimation of the sensor measures may not be interesting. In fact, it depends on the designer specifications. With respect to reconfiguration, faults on the electrical part of the system can be tolerated. Also the system can be reconfigured in the presence of $(\dot{\theta}_{sj})$ because it remains observable with the remaining healthy sensors.

Co-simulation validation:

To validate the structural results, co-simulations are presented for the first scenario. The results are performed on a vehicle dynamic simulator software (*SCANeR Studio*)[OKTAL 2012] that gives a three dimensional performance of the vehicle model in response to input controls. The latter is a validated automotive driving simulator, dedicated to engineering and research. In the following co-simulation results are produced using two external programs under independent supports, where data is exchanged between the two programs. In this case, we co-simulate between a program im-

plemented under *Matlab/Simulink* environment and *SCANeR Studio* simulator. Since RobuTAINeR still under development, for the following results, the whole dynamics of the RobuTAINeR were furnished by the *SCANeR Studio*, based on the vehicle characteristics obtained from RobuTAINeR developer's. So we use this platform to co-simulate the controller and the dynamics of the actuated electromechanical traction subsystem developed on *Matlab/Simulink* (Figure 4.10).



Figure 4.10: *SCANeR Studio* driving simulator and the wheel frame of RobuTAINeR that was modeled with *Matlab/Simulink*.

To perform the co-simulations, we considered a section of an existing course on the Radicatel terminal at the port of Rouen, Normandie - France, partner in InTraDE project. The trajectory in the Terminal of Radicatel is presented in Figure 8 (red line) and in 2D real coordinates. It is based on tracking a slightly curvy road until RobuTAINeR arrives to a roundabout. Then, the vehicle drives around the roundabout and exits it when a turn of approximately 360 degrees is completed. Finally, a straight line must be tracked.

The parameters of the RobuTAINeR are defined in Table 4.4.

The input fault is $Se : U_{01}$ introduced in the front left actuated subsystem at 50s, which triggers r_{11} , as depicted in Figure 4.12. The red dashed lines represent robust thresholds. These are obtained with respect to parameter [Djeziri 2007b] and measurement [Touati 2011] uncertainties. As previously explained, this fault can be tolerated by system reconfiguration. To validate the structural results, we have switched the control strategy, where only the rear wheels are actuated. The trajectory tracked by



Figure 4.11: Radicatel terminal with desired trajectory on SCANeR Studio software.

RobuTainer is given in Figure 4.13. It is concluded from this figure that the desired trajectory remains tracked, even after the fault occurrence.

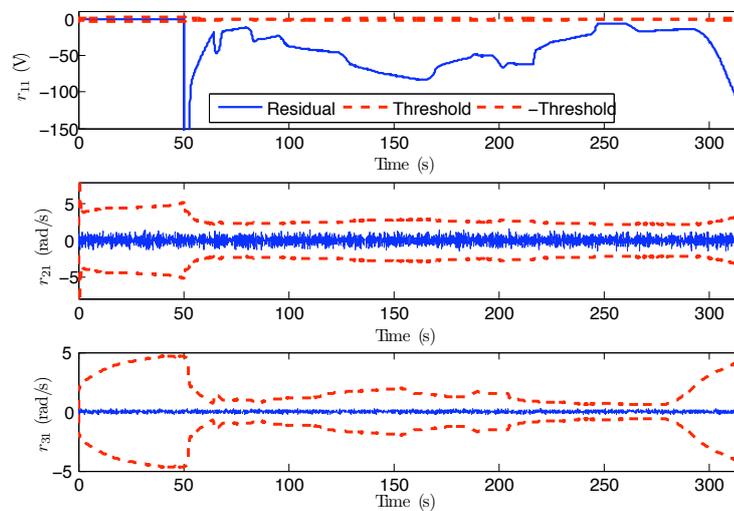


Figure 4.12: Residuals of the front left electromechanical subsystem, under input faulty conditions

4.3.3 Structural analysis for local adaptive compensation

It is clear from Table 4.3, that by the classical conditions for fault compensating with fault accommodation or system reconfiguration, a small set of faults can be tolerated.

Subsystem	Nominal values		
Electromechanical subsystem	L_{ej}	0.022	(H)
	R_{ej}	1.5	(Ω)
	k_{ej}	2.37	($N.m/A$)
	J_{ej}	0.00177	($Kg.m^2$)
	f_{ej}	0.3068	($N.m.sec.rad^{-1}$)
	J_{sj}	2	($Kg.m^2$)
	f_s	0.2	($N.m.sec.rad^{-1}$)
	N_j	0.1	–
	r	0.36	(m)
Vehicle body	m	2917.2	Kg
	J	1756.6	Kgm^2
	a	2.3	m
	b	2.3	m
	c	0.86	m

Table 4.4: System parameters

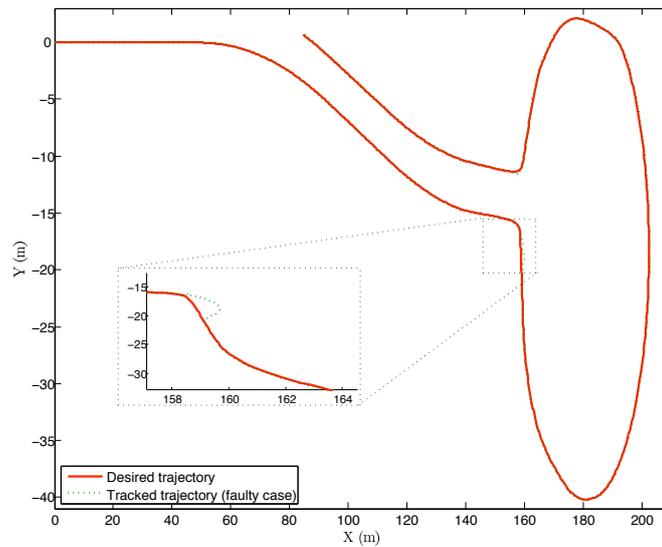


Figure 4.13: Desired trajectory (solid line) and tracking results when operating under input faulty (dotted line) conditions.

To this end, we proposed new structural conditions for a local adaptive fault compensation in Chapter 3. By applying these conditions into the j^{th} electromechanical subsystems, we can conclude from Table 4.3 that we have three isolated junctions with no detectors contained in their fault signatures, and the isolation of the element

$MGY : k_{ej}$. These junctions are the following:

$$\begin{cases} 1_1 - \text{junction} & : F_{sig}^b = \{[1, 0, 0, 1, 0]\} \\ Elements & : \{Se : U_{0j}, R : R_{ej}, I : I_{ej}\} \end{cases} \quad (4.32)$$

$$\begin{cases} 1_2 - \text{junction} & : F_{sig}^b = \{[0, 1, 0, 1, 1], \} \\ Elements & : \{R : f_{ej}, I : J_{ej}\} \end{cases} \quad (4.33)$$

$$\begin{cases} 1_3 - \text{junction} & : F_{sig}^b = \{[0, 0, 1, 0, 1], \} \\ Elements & : \{R : f_{sj}, I : J_{sj}, TF : r, MSe : Fl_j\} \end{cases} \quad (4.34)$$

To detail the obtained results, let us consider a fault in the mechanical resistance (f_{e1}) of the DC-motor. This element belongs to the 1_2 -junction and it is isolable. In this case, the BG-LFT formalism is applied to the front left electromechanical subsystem on the $R : f_{ej}$ element. Figure 4.14 shows the employed procedure. In part (a) of the figure, the BG-LFT model for fault estimation is performed. While in part (b) this estimation is furnished to the inverse BG-LFT model of the system.

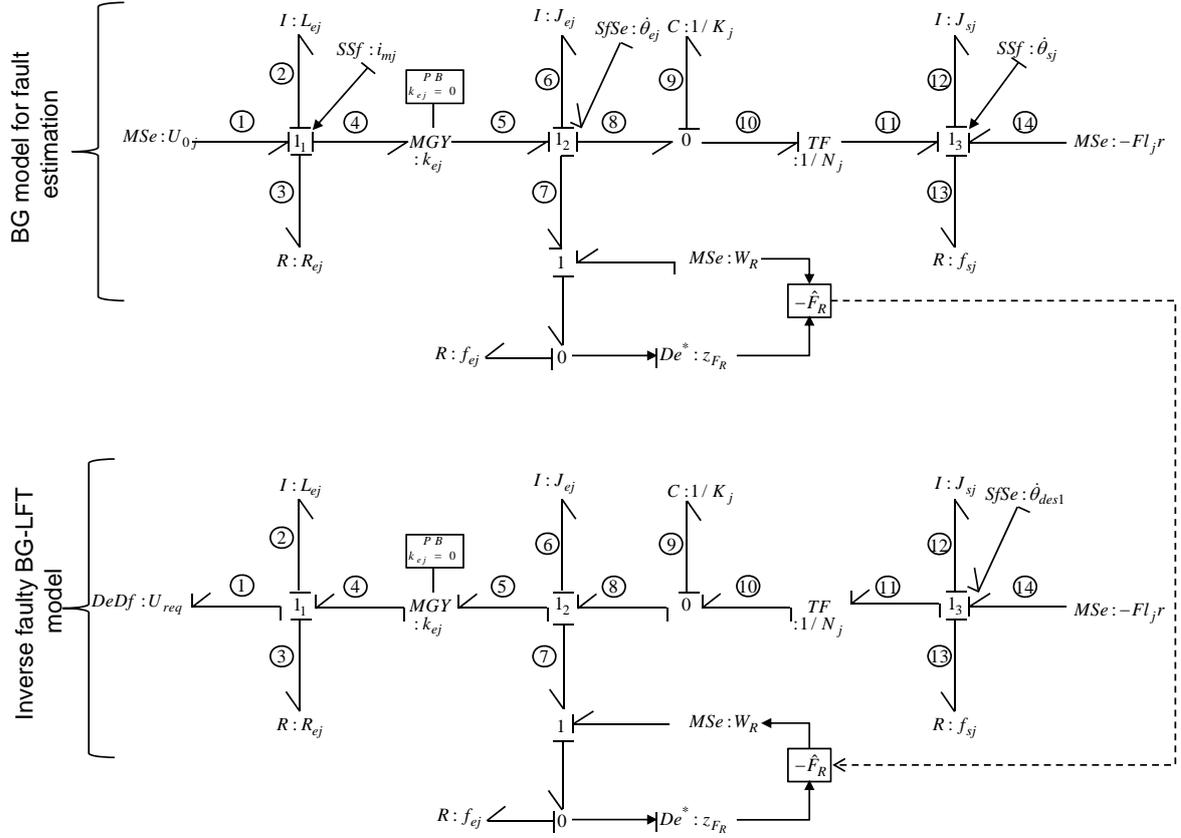


Figure 4.14: (a) BG-LFT for fault estimation, and (b) inverse BG-LFT for system inversion.

Co-simulation validation:

For validation purposes, this fault is introduced at 15sec . The desired angular velocity and the output of the inverse system is given in Figure 4.15.

The residuals are depicted in Figure 4.16, and 4.17. As expected, r_2 , r_4 , and r_5 are triggered. The red dashed lines represent robust thresholds. In this co-simulation, the thresholds are obtained with respect to parameter [Djeziri 2007b] uncertainties.

The estimation faulty value F_R (\hat{F}_R) is given in Figure 4.18. From Figure 4.15, one can remark that the voltage increases at the moment of the fault due to this injection. Finally, from Figure 4.18 we notice that the fault causes an errors on the tracking speed of approximately 0.15rad/s and that it is very quickly compensated.

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

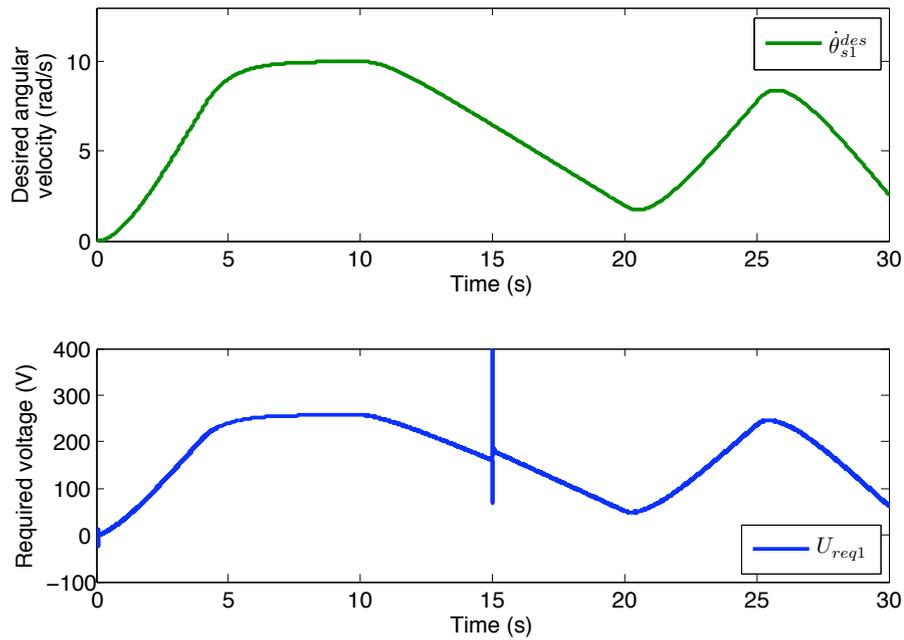


Figure 4.15: Desired velocity of RobuTAINeR’s wheels and required voltage obtained from system inversion.

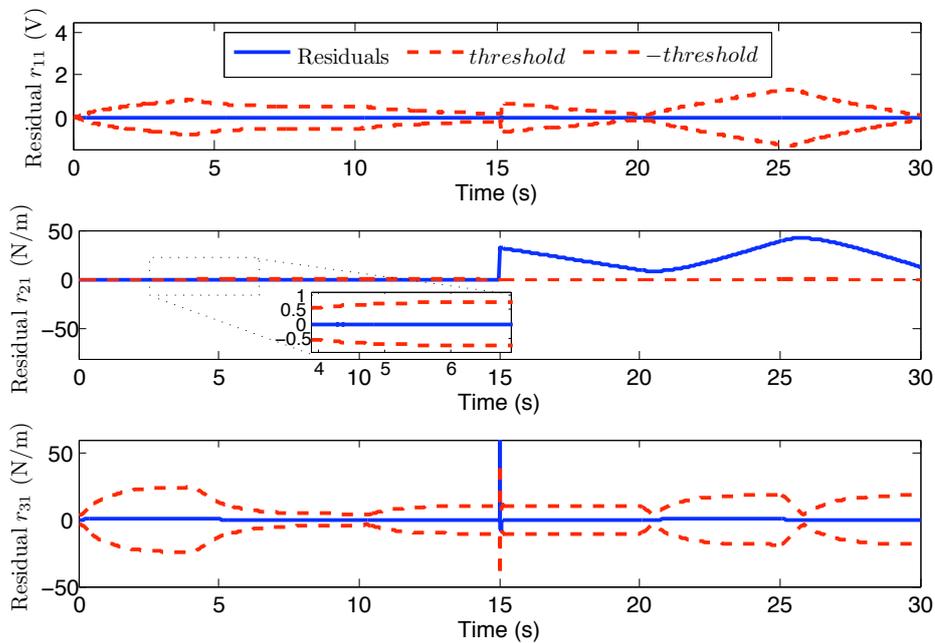


Figure 4.16: Residuals 1, 2, and 3 of the front left electromechanical subsystem.

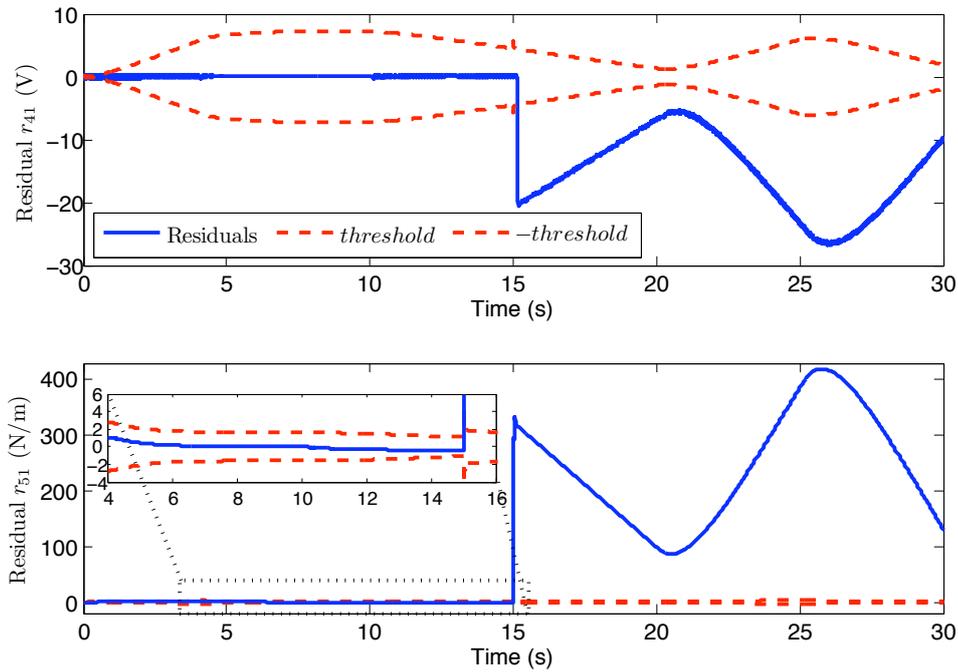


Figure 4.17: Residuals 4, and 5 of the front left electromechanical subsystem.

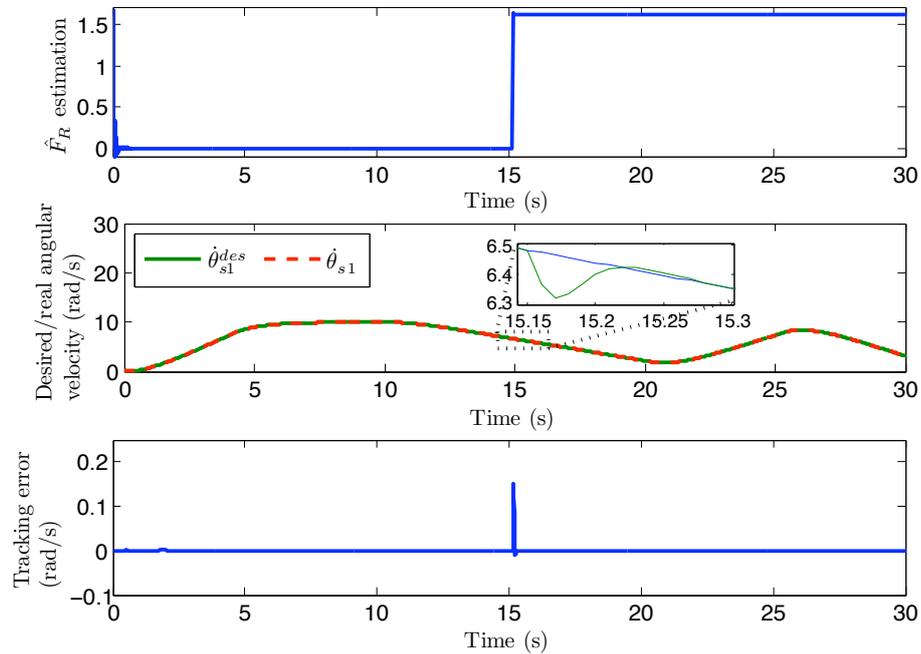


Figure 4.18: Fault estimation, tracked angular velocity, and tracking error.

4.4 Conclusions

In this chapter we detailed the procedure of structural fault tolerance on an intelligent autonomous vehicle. This is a MIMO overactuated complex system, however the

Case study: Co-simulations on a heavy size Intelligent Autonomous Vehicle

structural and causal properties of the BG tool enable to conclude about the fault tolerance properties previous to industrial implementation. It was concluded that in some situations, component fault isolability is not a necessary condition for system reconfigurability. Due to this fact the number of faults that a system can tolerate when proper control actions are applied may increase. Nevertheless, it was shown that the set of fault that could be tolerated are limited. Hence, structural conditions for a local adaptive compensation were tested and the number of faults that can be compensated increased substantially. The obtained structural results were validated through co-simulations.

Conclusions and recommendations for future research

Conclusions

In this thesis, we proposed a new approach for structural fault recoverability analysis on dynamic systems. This was done with the same graphical tool, the bond graph. Indeed, this tool enables to go from dynamic modeling, to structural analysis for diagnosis and finally for structural fault tolerance analysis.

The bond graph model of the system was obtained, which enables to deduce Analytical Redundancy Relations (*ARRs*) in a systematic way. In addition, in this work, an approach to generate additional *ARRs*, using the bicausal notion was proposed. These *ARRs* enable to obtain the different set of fault signatures, and then, this information was exploited to evaluate the level of the systems fault tolerance with respect to the different possible fault locations: actuator, sensor, and plant faults. To this end, structural conditions were furnished and a Fault Signature and Recoverability Matrix (*FSRM*) was proposed. From this matrix the set of faults that prevent the systems objectives to be achieved are concluded, which are referred to as critical faults. It is concluded that in some situations, component fault isolability is not a necessary condition for system recoverability. Due to this fact, the number of faults that a system can tolerate when proper control actions are applied may increase. The proposed procedure is a preliminary study before implementing a closed loop control. It allows from system design point of view, to study all conditions related to fault recoverability, inspired from the causal and structural properties of the bond graph model.

Nevertheless, most of the time, fault isolability is a main requirement for system recoverability. However, the diagnosis algorithm is often unable to isolate the exact faulty component of the system. Thus, the set of fault that can be tolerated are usually limited. To overcome this drawback, a novel way to consider the fault not as a information but as a faulty power was introduced. Based on this idea, structural conditions were given so that a local adaptive compensation can be performed. To this end, the procedure of Bond Graph in the Linear Fractional Transformation (BG-LFT) was used to estimate the faulty power. Then, to validate these structural results,

a local adaptive compensation based on the inverse control strategy using the inverse BG-LFT was proposed. This strategy computes the desired inputs based on the system objectives and on the undesired power caused by the fault.

Finally, to validate the obtained structural results, the developed procedures in the framework of this thesis were verified in an redundant over-actuated intelligent electric autonomous vehicle. We have described a simplified vehicle model of the vehicle, with four electromechanical subsystems for traction. Each traction subsystem is composed of three sensors, that are exploited for the generation of *ARRs*. The diagnosis analysis is then further used to evaluate the fault tolerance of the overall vehicle. Once the structural results were obtained, co-simulation were performed in order to validate them.

Thus, the work proposed in this thesis has extended the works initially developed at the group *MOCIS*, to the concept of fault tolerance.

Recommendations for future research

We have performed the fault tolerance analysis from a structural point of view. We believe that studies regarding sensor and actuator placement with respect to fault tolerance measures could be an interesting extension of this work. Moreover, since the bond graph explicitly models the energy propagation in the system. In future research, fault tolerance could be extended for online analysis, where the energy required to control the system until the end of its mission is evaluated.

In addition, further analysis with a larger set of faults for the adaptive fault compensation on the control by BG-LFT inverse should be performed. The concept of the power caused by the fault with a more powerful control law, which uses the model of the system could also be considered. Indeed, if a source representing the faulty power is introduced in the system and/or constraints, the control problem could be easily modified.

Conclusions

Appendix A

Bond Graph generalities

A.1 Bond graph modeling

The BG is a unified modeling language for several physical domains, where the power variables associated with the bonds differ from the physical domain of the model. Hence, the type of energy exchanged is defined by the power variables. A non exhaustive description of these variables is displayed in Table A.1.

Table A.1: Effort and flow variables in different physical domains

Physical domain	effort (e)	flow(f)
Mechanical (Translational)	Force (F)	Velocity (v)
Mechanical (Rotational)	Torque (T)	Angular velocity (ω)
Electrical	Voltage (V)	Current (i)
Hydraulic	Pressure (P)	Volume ow rate (Q)
Thermal	Temperature (T)	Entropy flow rate (\dot{s})

Moreover, the BG tool divides the physical component into Active, Passive, and multi-port elements, as detailed under.

Active elements:

Active elements are sources that supply power to the system (Sources of effort (Se), and flow (Sf)). The bond orientation always goes out of the source.

*Source of effort-(Se):*Gives an effort to the system $e = Se$ (see Figure A.1-(a)). An

Bond Graph generalities

example of physical components modeled by an Se element is a voltage supply.

Source of flow-(Sf): Gives a flow to the system $f = Sf$ (see Figure A.1-(b)). For instance, physical components modeled by an Se element.

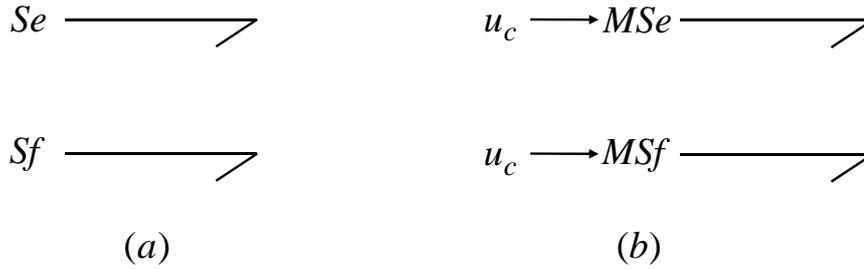


Figure A.1: Active elements: Sources of effort and flow (a), modulated sources of effort and flow (b).

Passive elements

Passive or 1-port elements, are elements that transform the received power into dissipated power (R), stores as potential energy (C) or as kinetic (I) energy. The bond usually is directed onto these elements.

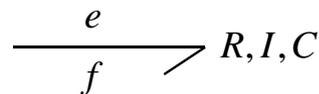


Figure A.2: Representation of 1- port passive elements: R -element (a), I -element (b), and C -element.

Resistor element-(R) is used to represent physical components that dissipate energy. Its correspondent constitutive relation is a static function ($\Phi_R(e(t), f(t)) = 0$) for Figure A.2-(a). Moreover, the latter can take the following two forms depending on the causality associated to the corresponding element $e(t) = R.f(t)$, $f(t) = \frac{1}{R}.e(t)$. Some examples of physical components modeled by an R element are: electrical resistors, mechanical dampers, and hydraulic valves.

Inertial element-(I) is used to model the energy storage phenomena that is defined by a constitutive equation relating the flow and the integral of effort ($\Phi_I(f(t), \int e(t)dt) =$

0) for Figure A.2-(b). The latter may take the form $f(t) = \frac{1}{I} \int e(t)dt$ or $e(t) = I \frac{d}{dt} f(t)$, depending on its causality. Some examples of physical components modeled by an I element are: electrical inductance, mass, inertial components.

Capacitor element-(C) is used to model the energy storage phenomena that is defined by a constitutive equation relating the effort and the integral flow ($\Phi_I(e(t), \int f(t) dt) = 0$) for Figure A.2-(c). The latter equation can take the form $e(t) = \frac{1}{C} \int f(t)dt$ or $f(t) = C \frac{d}{dt} e(t)$, depending on its causality. Some examples physical components modeled by an C element are: electrical capacitance, springs, tanks. From these elements, we deduce the set of behavioral equations (C_B) as follows:

$$\begin{aligned}
 C_B &= \{C_R\} \cup \{C_I\} \cup \{C_C\}, \\
 C_R &= \Phi_R(f_R(t), e_R(t)) = 0, \\
 C_I &= \Phi_I\left(f_I(t), \int e_I(t)dt\right) = 0, \\
 C_C &= \Phi_C\left(e_C(t), \int f_C(t)dt\right) = 0.
 \end{aligned} \tag{A.1}$$

Junction elements

Junction elements are used when energy is transferred between two subsystem, or to represent the structure of the system. The latter is modeled by TF and GY elements while the former used 1 - and 0 - junctions.

Transformer element-(TF) is used to connect two subsystems when the output effort (flow) of one subsystem is not equal to the input effort (flow) of the other subsystem (see Figure A.4-(a)). It can be used to describe connections between different types of physical variables. The constitutive equations associated to a TF are: $e_1 = m e_2$, and $f_2 = m f_1$. Some examples of physical components modeled by a TF element are: hydraulic cylinders, gear pairs, wheels. *Gyrator element-(GY)* is used to connect two subsystems when the output effort (input flow) of one subsystem is proportional to the input flow (output effort) of the other subsystem (see Figure A.4-(b)). The constitutive equations associated to a GY are: $e_1 = k f_2$, and $e_2 = k f_1$. Some examples physical

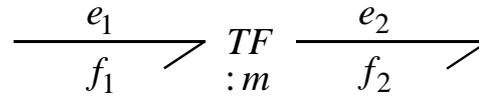


Figure A.3: 2-port *TF*-element.

components modeled with a *TF* element are: gyroscopes, electric motors.

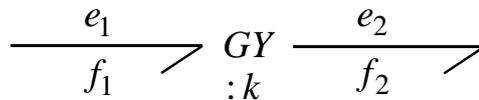


Figure A.4: 2-port *GY*-element.

0 (1)-junction is used when the elements connected to it have a common effort (flow). The equations deduced from junction are named *structural equation*. From the 0 -junction illustrated in Figure A.5, the structural equations are the following:

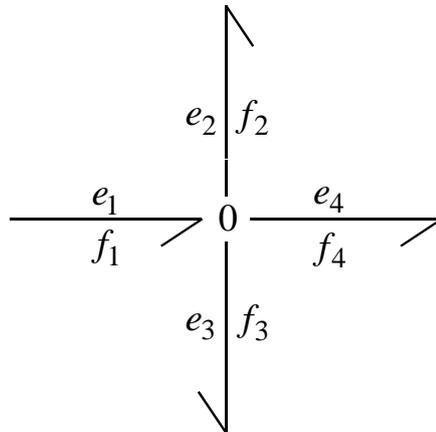


Figure A.5: BG 0 -junction.

$$\begin{aligned}
 e_1 &= e_2 = e_3 = e_4, \\
 \sum f_i &= 0; f_1 - f_2 - f_3 - f_4 = 0.
 \end{aligned}
 \tag{A.2}$$

Note that the power bonds pointing to the junction have a (+) signal associated to its flow while the others have a (-) signal. Moreover, from the 1 -junction illustrated in Figure A.6, the structural equations are the following:

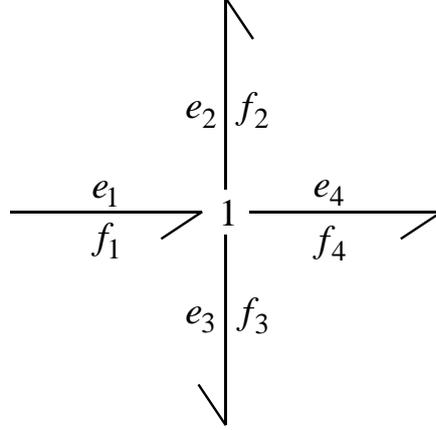


Figure A.6: BG 1-junction.

$$\begin{aligned}
 f_1 = f_2 = f_3 = f_4 = \dots = f_n, \\
 \sum e_i = 0; \quad e_1 - e_2 - e_3 - e_4 = 0.
 \end{aligned} \tag{A.3}$$

Finally, the set of structural constraints (C_S) are the following:

$$\begin{aligned}
 C_S &= \{C_{J_0}\} \cup \{C_{J_1}\} \cup \{C_{GY}\} \cup \{C_{TF}\}, \\
 C_{J_0} &: \Phi(\sum f_i) = 0, \\
 C_{J_1} &: \Phi(\sum e_i) = 0, \\
 C_{GY} &: \Phi_{f_{GY}} = (f_{1_{GY}}, e_{2_{GY}}) = 0 \quad \text{and} \quad \Phi_{e_{GY}} = (e_{1_{GY}}, f_{2_{GY}}) = 0, \\
 C_{TF} &: \Phi_{f_{TF}} = (f_{1_{TF}}, f_{2_{TF}}) = 0 \quad \text{and} \quad \Phi_{e_{TF}} = (e_{1_{TF}}, e_{2_{TF}}) = 0.
 \end{aligned} \tag{A.4}$$

Finally, to completely represent the system architecture the measurement set should also be defined. Hence, a detector of effort (De), and of flow (Df) are added. Power bonds are not used to connect the junctions with detectors because there is no transfer of power between them. Hence, a full arrow representing an information bond is used. As previously referred elements associated to a 1-junction have common flow, hence a detector Df is the one used at these junctions, and they measure power variables such as current, velocity, etc. Consequently, detectors De are connected to θ -junctions, and they measure power variables such as voltage, pressure, etc. Some examples of De are:

voltmeter, sensor of pressure. For Df one can exemplify an ampere-meter or a flow rate sensor. The set of measurement constraints are defined as follows:

$$\begin{array}{ccc} \xrightarrow[e=0]{e} De & \xrightarrow[f]{e=0} Df \\ (a) & (b) \end{array}$$

Figure A.7: (a), (b) representation of a detector of effort and flow, respectively.

$$\begin{aligned} C_M &= \{C_{m_f}\} \cup \{C_{m_e}\}, \\ C_{m_f} &: \Phi m_f = (Df, f_J), \\ C_{m_e} &: \Phi m_e = (De, e_J). \end{aligned} \tag{A.5}$$

A.2 Causality

An important property of the bond graph is the causality. The latter enables to define the cause-effect relations in a system. The type of causality used in a model is related to the causality assigned to the storage elements I and C . Indeed, the causality assigned to these elements determine if either an integration or a differentiation with respect to time is required. For the storage elements the causal strokes in preferred integral causality are assigned as illustrated in Figure A.8-(a). Computationally it means that the inertia element accepts an effort as input as produces a flow as output, while the capacitor accepts flow as input and produces effort (A.6),

$$(a) : \begin{cases} f_I(t) = \frac{1}{I} \int e_I(t) dt, \\ e_C(t) = \frac{1}{C} \int f_C(t) dt. \end{cases} \tag{A.6}$$

If a derivative causality is assigned, as in Figure A.8-(b), the I elements accepts a flow as input and produces an effort as output, while the C element accepts effort as

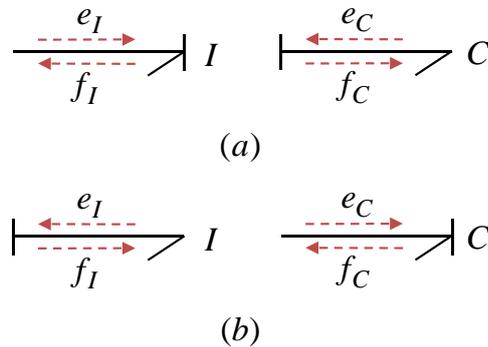


Figure A.8: (a) Storage elements I and C in integral causality, (b) Storage elements I and C in derivative causality.

input and produces flow as output (A.7).

$$(b) : \begin{cases} e_I(t) = I \frac{df_I(t)}{dt}, \\ f_C(t) = C \frac{de_C(t)}{dt}. \end{cases} \quad (\text{A.7})$$

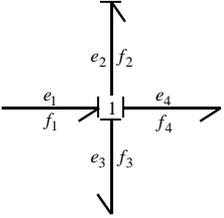
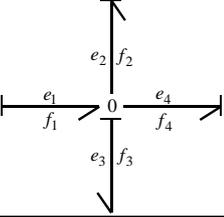
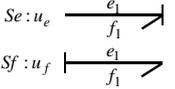
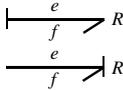
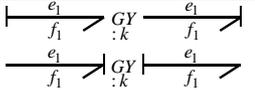
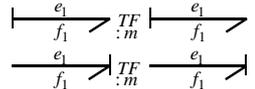
The remaining BG elements with its rules of causality are detailed in Table A.2

The procedure of causality assignment in a BG model is named as: Sequential Causality Assigned Procedure (SCAP).

Sequential causality assignment procedure (SCAP) [Samantaray 2008b]

1. The sources always impose one causality. Choose any source and assign its required causality. Extend the causal implications with respect to the rules of causal assignment in $0, 1, GY, TF$.
2. Choose any C - or I -element and assign integral causality. Extend the causal implications with respect to the rules of causal assignment in $0, 1, GY, TF$.
3. Choose any R -element that is unassigned and give it an arbitrary causality. Again, extend the causal implications with respect to the rules of causal assignment in $0, 1, GY, TF$.

Table A.2: Causality of BG elements.

Bond graph	Causal equation	Rule
	$\begin{cases} f_1 = f_2 = f_3 = f_4 \\ e_2 = e_1 - e_3 - e_4 \end{cases}$	<p>Only one flow imposed onto the 1-junction</p> <p>Rule: Only one stroke far from the 1-junction).</p>
	$\begin{cases} e_1 = e_2 = e_3 = e_4 \\ f_3 = f_1 - f_2 - f_4 \end{cases}$	<p>Only one effort imposed onto the 0-junction</p> <p>Rule: Only one stroke near the 0-junction).</p>
	$e_1 = u_e$ $f_1 = u_f$	<p>Effort (flow) source imposes effort (flow) onto the junction.</p> <p>Rule: Compulsory causality.</p>
	$e = Rf$ $f = e/R$	<p>- Resistive causality</p> <p>- Conductive causality</p>
	$e_1 = kf_2, e_2 = kf_1$ $f_2 = e_1/k, f_1 = e_2/k,$	<p>Two flows or two efforts imposed onto GY</p>
	$e_1 = me_2, f_2 = mf_1$ $e_2 = e_1/m, f_1 = f_2/m,$	<p>One flows and one efforts imposed onto TF.</p>

A.3 State space equations associated to a bond graph model

The system state space equations can be derived from a bond graph model by introducing the constitutive equations for each subsystem (behavioral equations) and the constraints imposed by the junctions (conservation law equations). The dimension of the state vector is equal to the number of I and C elements in integral causality. Moreover, the state vector of a system (x) is composed of energy variables p and q

associated to the I , and C , elements, respectively.

$$x = \begin{bmatrix} p_I \\ q_C \end{bmatrix} = \begin{bmatrix} \int e_I \\ \int f_C \end{bmatrix}. \quad (\text{A.8})$$

The state variables is not presented in the BG model, only its derivative.

$$\dot{x} = \begin{bmatrix} \dot{p}_I \\ \dot{q}_C \end{bmatrix} = \begin{bmatrix} e_I \\ f_C \end{bmatrix}. \quad (\text{A.9})$$

A.4 Bicausal assignment procedure

In a bicausal junction, only one bond brings in both effort and flow information and another bond takes out both flow and effort information. Therefore, only two bonds can be put in bicausality in each junction [Gawthrop 1995, Samantaray 2008b]. More formally, in a θ -junction the following two rules must be respected for a bicausal assignment (Figure A.9-(a)):

1. Exactly one effort is impinged on a zero junction with N bonds attached.
2. Exactly $N-1$ flows are impinged on a zero junction with N bonds attached.

For 1 -junctions, the procedure of bicausal assignment is analyzed in a similar fashion (Figure A.9-(b)). The bicausal assignment on the θ -junction of Figure A.9-a implies

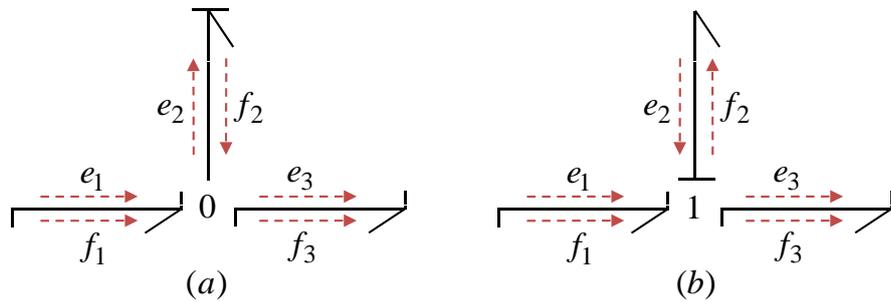


Figure A.9: (a) bicausal assignment in a θ -junction, (b) bicausal assignment in a 1 -junction.

that f_3 and e_3 are the information that must be computed in this junction ($f_3 = f_1 - f_2$). Similarly, for the 1 -junction of Figure A.9-a the bicausality imposes $e_3 = e_1 - e_2$.

A.5 Bond graph LFT for fault estimation

The bond graph in linear fractional transformation (BG-LFT) is used to model the fault in an elegant way directly on the bond graph model. To this end, let us introduce the concept of BG-LFT for fault modeling.

A.5.1 Linear fractional transformation (BG-LFT) for fault modeling

The concept of bond graph in linear fractional transformation, was initially introduced in [Kam 2005] to model parameter uncertainties. Then, this concept was exploited in [Djeziri 2007b] for robust fault diagnosis and, more recently, [Touati 2012] used it for fault modeling and estimation.

To explain the idea of BG-LFT for fault modeling, let us consider a multiplicative fault on an R -element in resistive causality as follows:

$$e_F = R_n(1 + F_R)f_R. \quad (\text{A.10})$$

Where, e_F is the faulty effort, R_n is the resistive element value under nominal conditions, and F_R is the value of the fault on R_n . The corresponding block diagram of equation (A.10) is depicted in Figure A.10. This is easily associated to the BG-LFT model as in Figure A.11.

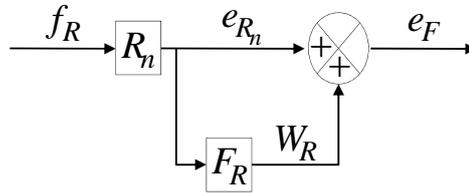


Figure A.10: Block diagram LFT of an R-element in resistive causality with a multiplicative fault.

Remark. The (-) sign appearing in the BG model of Figure A.11, is caused by the signal convention of the junction 1 and 0. The symbols De^* and Df^* are used to

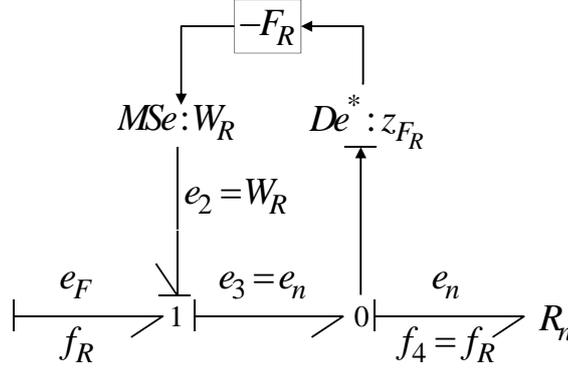


Figure A.11: BG model LFT of an R-element in resistive causality with a multiplicative fault.

represent the fact that they are virtual detectors.

In the BG model of Figure A.11, the virtual detector $De^* : z_{FR}$ are introduced to explain the logic of BG-LFT but this variables are known (system is assumed observable). Moreover the modulated source $MSe : W_R$ is added to represent the introduction of an additional effort generated by the fault (W_R) on the system.

This procedure can be extended to model three different types of faults, namely: parameters, actuators, and sensors. In a BG model, parameters faults are associated with an unexpected variation of the basic element R, I, C, GY, TF . Faults on one-port elements are modeled with the BG-LFT, as depicted in Figure A.12.

By covering the path on the model presented in Figure A.12-(d), the following equation are obtained:

$$1 - \text{junction} : \begin{cases} f_1 = f_2 = f_3, \\ e_1 = e_F = e_3 - e_2. \end{cases} \quad (\text{A.11}) \quad 0 - \text{junction} : \begin{cases} f_3 = f_4, \\ e_3 = e_4 = e_n. \end{cases} \quad (\text{A.12})$$

$$R - \text{element} : \begin{cases} e_4 = R_n \cdot f_4. \end{cases} \quad (\text{A.13})$$

$$LFT - \text{procedure} : \begin{cases} z_{FR} = e_5 = e_n, \\ W_R = -F_R \cdot z_{FR}. \end{cases} \quad (\text{A.14})$$

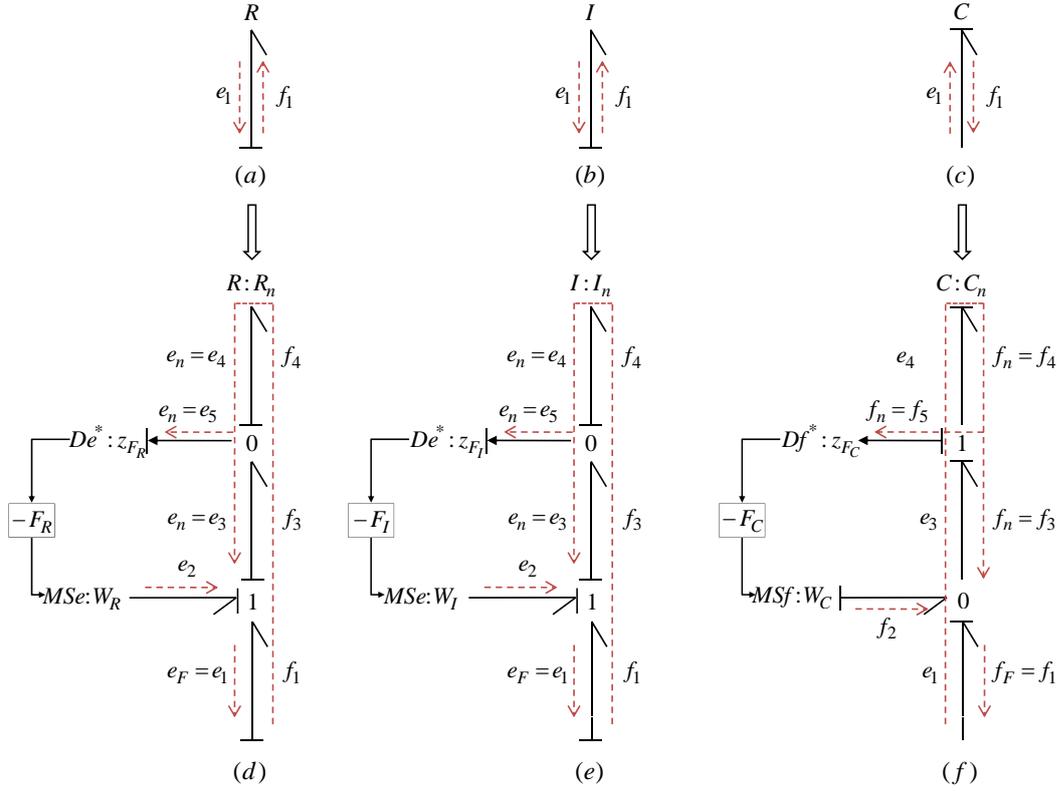


Figure A.12: BG-LFT for parameter fault modeling.

Therefore, the faulty effort e_F is obtained (A.15).

$$e_F = R_n(1 + F_R)f_1. \quad (\text{A.15})$$

The multiplicative faults on the dynamic elements I and C are also represented in the same way as on an R element, as presented in Figure A.12-(e) and (f). In this case, F represents the function connecting Z and W by considering the fault variation. For instance, on a I element in derivative causality, the following equations can be obtained:

$$f_n = \frac{d((C_n + C_n \xi_C) e_4)}{dt} = C_n \frac{de_4}{dt} + \underbrace{C_n \xi_C \frac{de_4}{dt} + C_n e_4 \frac{d\xi_C}{dt}}_{W_C}$$

$$f_n = \frac{d((C_n + C_n \xi_C) e_4)}{dt} = C_n \frac{de_4}{dt} + \underbrace{\xi_C Z_c + \frac{d\xi_C}{dt} \int Z_c dt}_{W_C}$$

$C_n \xi_C$ is the instantaneous variation on the element C_n .

If the R_n element is in conductive causality a slightly different modeling procedure must be considered (Figure A.13). From this graph, the following equations are ob-

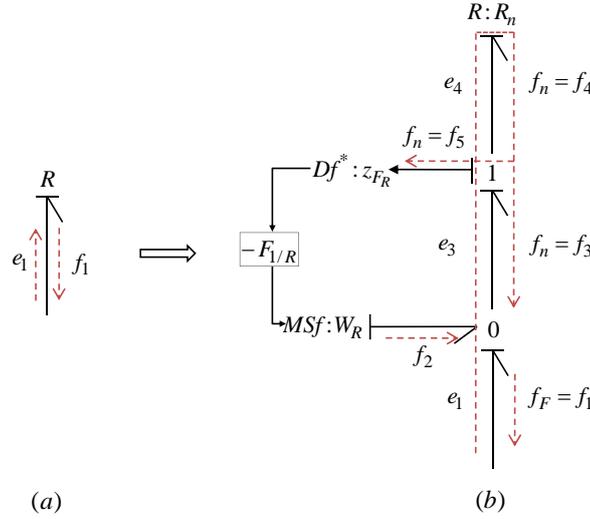


Figure A.13: BG-LFT for R -parameter fault modeling in conductive causality.

tained:

$$0\text{-junction} : \begin{cases} e_1 = e_2 = e_3, \\ f_1 = f_F = f_3 - f_2. \end{cases} \quad (\text{A.16}) \quad 1\text{-junction} : \begin{cases} e_3 = e_4, \\ f_3 = f_4 = f_n. \end{cases} \quad (\text{A.17})$$

$$R\text{-element} : \begin{cases} f_4 = \frac{e_4}{R_n}. \end{cases} \quad (\text{A.18})$$

$$LFT\text{-procedure} : \begin{cases} z_{FR} = f_5 = f_n, \\ W_R = -F_{1/R} \cdot z_{FR}. \end{cases} \quad (\text{A.19})$$

Where, $F_{1/R} = -\frac{F_R}{1+F_R}$.

Modeling of faults on gyrators and transformer elements are represented in Figure

A.14 and A.15, respectively.

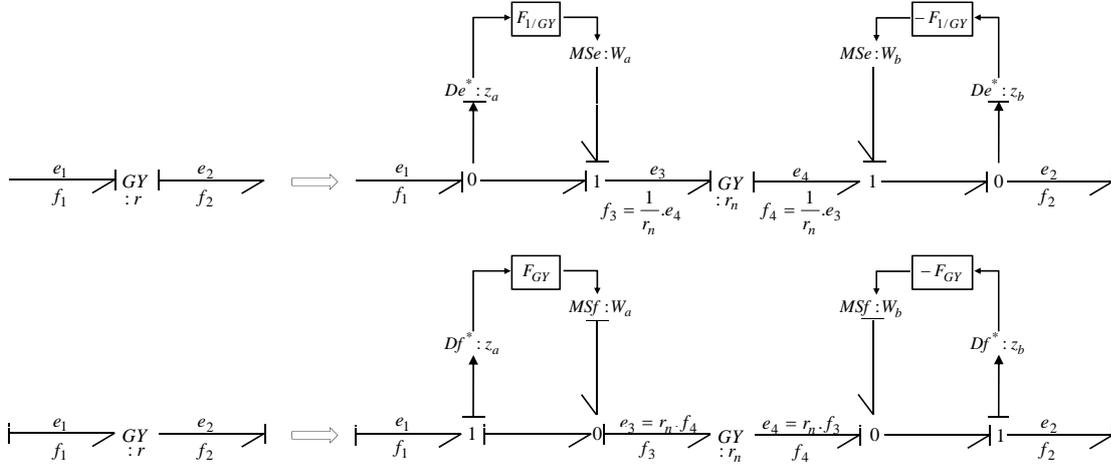


Figure A.14: BG-LFT for fault modeling on GY elements.

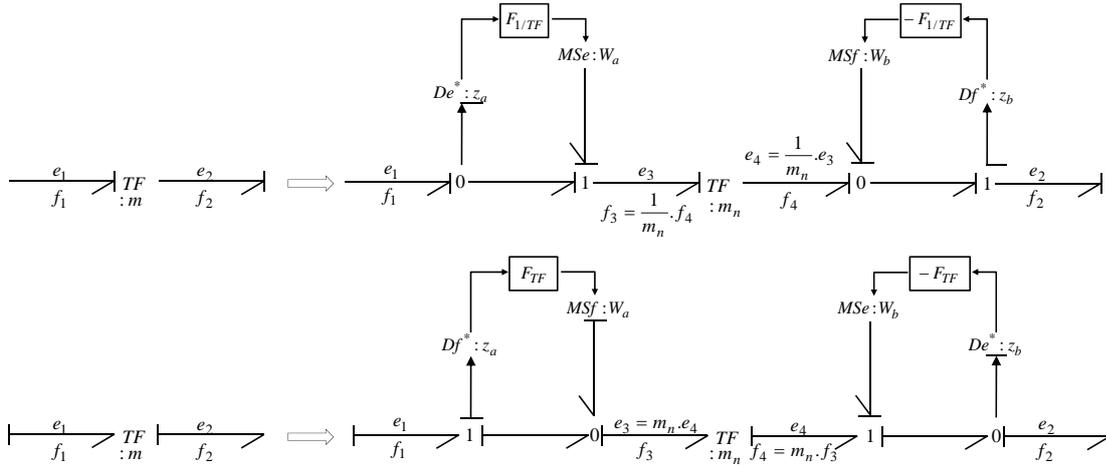


Figure A.15: BG-LFT for fault modeling on TF elements.

A.5.1.1 Input fault modeling

An actuator is said to be faulty if its output is different from the controller one (e_{ct} , f_{ct}). For modulated sources of flow or of effort we can model the fault on a BG model by adding a modulated source of effort ($MSe : F_{MSe}$) or of flow ($MSf : F_{MSf}$), depending on the nature of the faulty actuator (Figure A.16-(a), (b) for MSe , and MSf faults, respectively). Where F_{MSe} is the effort source fault value, while e_F is the real output of

the actuator. Similarly, F_{MSf} is the flow source fault value, and f_F is the real output of the actuator. From the BG representation one can write the following (A.20)-(A.21):

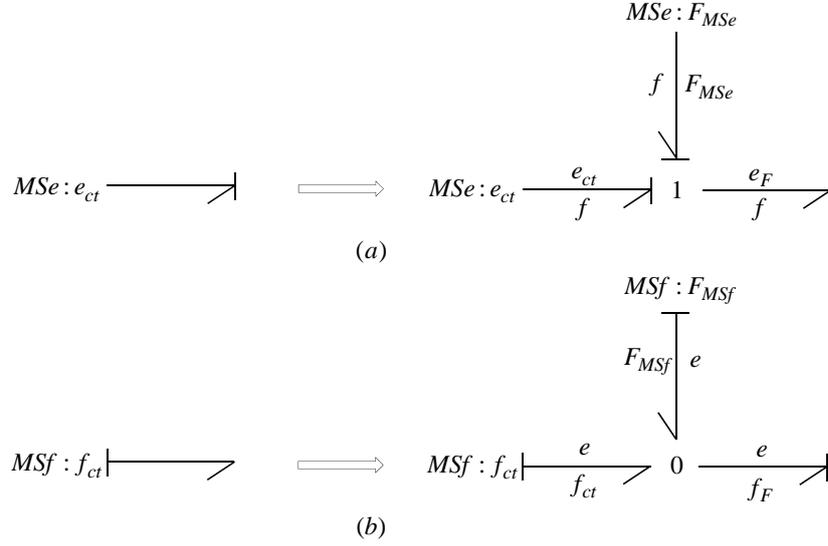


Figure A.16: BG-LFT for actuator fault modeling

$$MSe \rightarrow e_F = e_{ct} + F_{MSe}, \quad (\text{A.20})$$

$$MSf \rightarrow f_F = f_{ct} + F_{MSf}. \quad (\text{A.21})$$

A.5.1.2 Sensor fault modeling

A sensor is said to be faulty if its measured variable is different from the real one (e_r , f_r). To model faults on detectors of flow or of effort in a BG model, virtual modulated source of effort (MSe^*) or flow (MSf^*) are included on each bond connected to the junction associated to the faulty detector. The virtual source is used because a fault on a detector does not change the power in the system. Actually, it modifies the information used to calculate the power. From the BG representation depicted in Figure A.17, we can write the following (A.22)-(A.23):

$$SSe \rightarrow e_r = e_F - F_{SSe}, \quad (\text{A.22})$$

$$SSf \rightarrow f_r = f_f - F_{SSf}. \quad (\text{A.23})$$

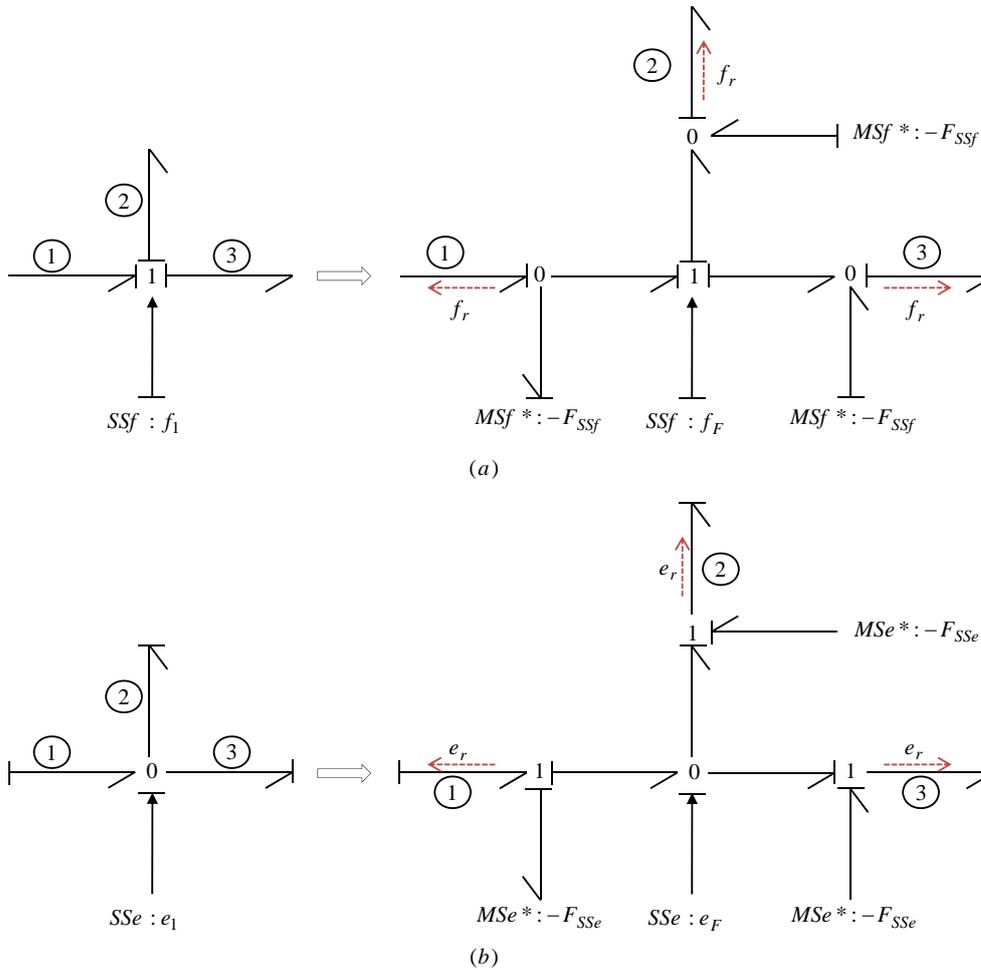


Figure A.17: BG-LFT for sensor fault modeling: (a) sensor of flow, (b) sensor of effort

A.5.2 Methodology for fault estimation

The modeling of the fault with the BG-LFT aims to parameters, sources, and detectors fault estimation. To this end, the notion of bicausality is exploited. Firstly, to be able to estimate the faults, the following two conditions must be respected [Touati 2012]:

1. The system is over-constrained,
2. The fault is isolable,
3. There is a causal path between the modulated input source representing the fault and a dualized detector.

A.5.2.1 Fault parameter estimation

For parameters fault modeling, the procedure is to apply the bicausality propagating from a detector to the modulated source of effort or flow caused by the fault. This procedure is illustrated in Figure A.18, for a fault in the R element, where the bicausality is propagated from $MSe : W_R$ to $SSf : f$. The BG-LFT is created for fault modeling (Figure A.18-(a)) and then, a bicausality between the detector (SSf) and the modulated source of effort representing the faulty effort ($MSe : W_R$) is assigned Figure A.18-(b). This graphical representation enables to obtain the following expres-

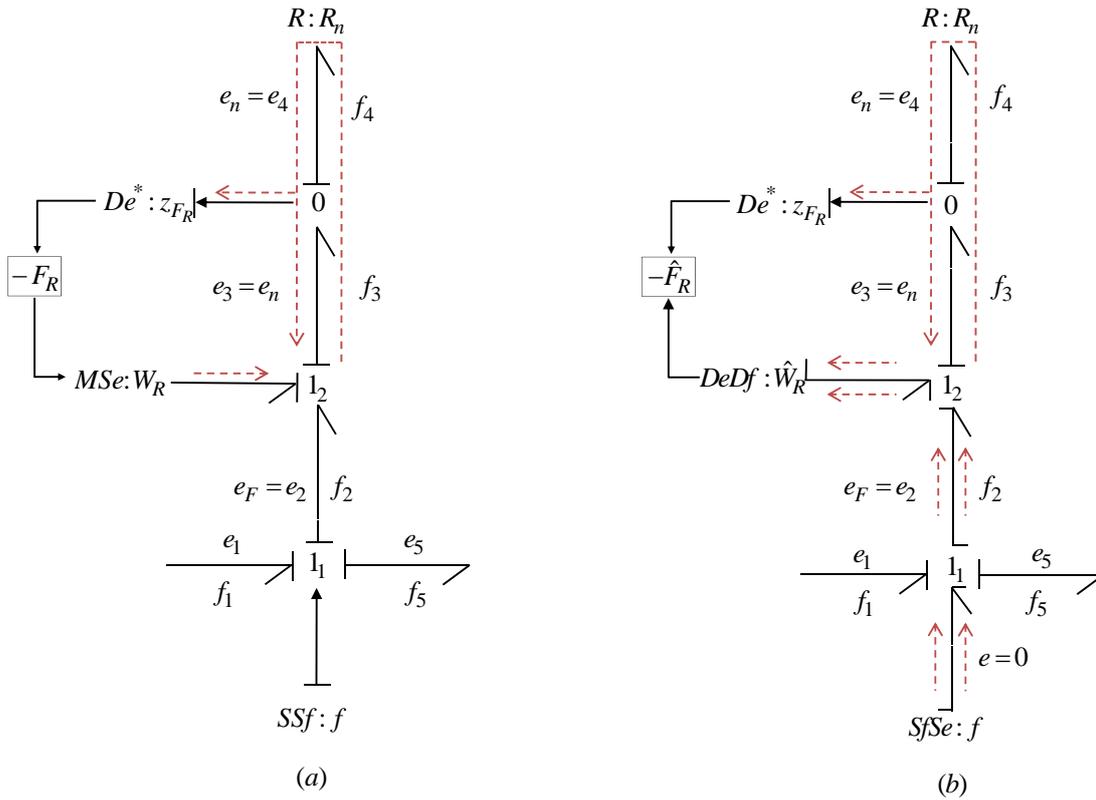


Figure A.18: (a) BG-LFT for R -parameter fault modeling, (b) Bicausal BG-LFT for R -parameters fault estimation.

sion for fault estimation in a systematic manner directly from the bicausal BG-LFT model presented in Figure A.18-(b).

$$1_1\text{-junction} : \begin{cases} f & = f_1 = f_2 = f_5, \\ e_2 & = e_F = e_1 - e_5. \end{cases} \quad (\text{A.24}) \quad 1_2\text{-junction} : \begin{cases} \hat{W}_R & = e_3 - e_2, \\ f_2 & = f_3. \end{cases} \quad (\text{A.25})$$

$$0\text{-junction} : \begin{cases} e_4 & = e_3 = e_n, \\ f_4 & = f_3. \end{cases} \quad (\text{A.26}) \quad R\text{-element} : \begin{cases} e_4 & = R_n \cdot f_4. \end{cases} \quad (\text{A.27})$$

$$LFT\text{-procedure} : \begin{cases} z_{FR} & = e_n, \\ \hat{F}_R & = -\frac{\hat{W}_R}{z_{FR}} = -\frac{e_3 - e_2}{R_n \cdot f_4}. \end{cases} \quad (\text{A.28})$$

A.5.2.2 Input fault estimation

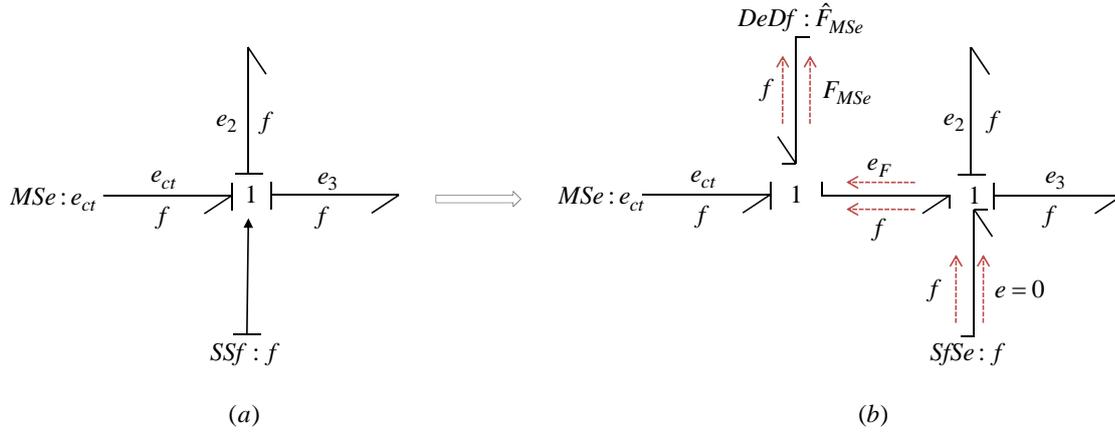


Figure A.19: (a) BG-LFT for *MSE* fault modeling, (b) Bicausal BG-LFT for *MSE* fault estimation.

If a fault in the modulated source *MSE* is considered, the bicausality is propagated from the modulated source ($MSe : F_{MSe}$) representing the fault to a detector ($SSf : f$). Then, the following expressions are obtained in a systematical manner directly from the bicausal BG-LFT model presented in Figure A.19-(b).

$$\hat{F}_{MSe} = -e_{ct} + e_2 + e_3. \quad (\text{A.29})$$

A.5.2.3 Sensor fault estimation

Finally, for sensor fault estimation, one must select one of modulated virtual sources representing the fault and propagate the bicausality to a dualized detector. Then, the faulty equation is generated by covering the path. This procedure is illustrated in Figure A.20.

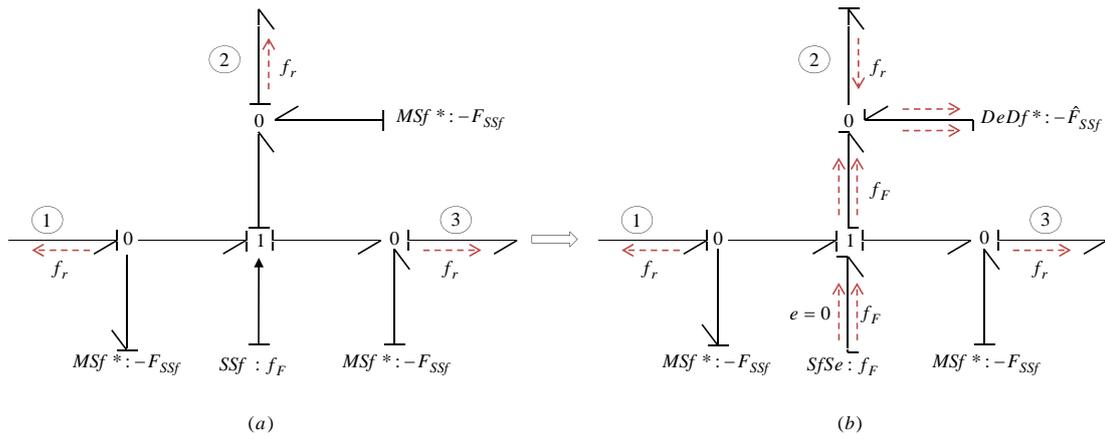


Figure A.20: (a) BG-LFT for Df fault modeling, (b) Bicausal BG-LFT for Df fault estimation.

Bibliography

- [Aitouche 2005] A. Aitouche and B. Ould-Bouamama. *Fault-Tolerant Control with respect to actuator failures application to steam generator process*. In Proceedings European Symposium on Computer Aided Process Engineering - 15, pages 1471–1476, 2005. 29, 31
- [Angeli 2004] C. Angeli and A. Chatzinikolaou. *On-line Fault Detection Techniques for Technical Systems: A survey*. International Journal of Computers Science & Applications, vol. 1, no. 1, pages 12–30, 2004. 15
- [Bartlett 2005] L. M. Bartlett and J. D. Andrews. *Fault tree based approach for system fault diagnostics*. In Proceedings of the 23rd International System Safety Conference, August 2005. 18
- [Blanke 2003] M. Blanke, M. Kinnaert, J. Lunze and M. Staroswiecki. *Diagnosis and fault-tolerant control*. Springer, 2003. 14, 18, 27, 38, 39, 40, 75
- [Boukhobza 2007] T. Boukhobza, F. Hamelin and S. Martinez-Martinez. *State and input observability for structured linear systems: A graph-theoretic approach*. Automatica, vol. 43, no. 7, pages 1204 – 1210, 2007. 39
- [Chang 1999] C. C. Chang and C. C. Yu. *On-Line Fault Diagnosis Using the Signed Directed Graph*. Industrial & Engineering Chemical Research., vol. 29, pages 1290–1299, 1999. 18
- [Chen 1999] J. Chen and R. J. Patton. *Robust model-based fault diagnosis for dynamic systems*. Kluwer Academic Publishers, Norwell, MA, USA, 1999. 15
- [Chow 1984] E. Y. Chow and A. S. Willsky. *Analytical Redundancy and the Design of Robust Failure Detection Systems*. IEEE Transactions on Automatic control, vol. 29, pages 603–614, 1984. 18

BIBLIOGRAPHY

- [Contreras 2011] B. M. G. Contreras, N. D. Checa, M. A. C. Aguilar, B. C. Michcol and F. E. M. Lopez. *Using a Realization Technique for System Reconfigurability Evaluation: Simulation Application on a DC Motor*. In Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2011 IEEE, pages 277–282, november 2011. 29, 31, 32
- [Dangoumau 2005] Nathalie Dangoumau and Armand Toguyéni. *System reconfigurability analysis in a decision support system for chemical manufacturing*. Discrete Event Dynamic Systems - DEEDS, vol. 19, no. 2, 2005. 29, 31, 32
- [de la Mata 2010] J. L. de la Mata and M. Rodríguez. *Accident prevention by control system reconfiguration*. Computers and Chemical Engineering, vol. 34, pages 846–855, 2010. 29, 31, 32
- [Demirci 2005] U. Demirci and F. Kerestecioglu. *Fault Tolerant Control With Re-Configuring Sliding-Mode Schemes*. Turk J Elec Engin, vol. 13, pages 175–187, 2005. 13
- [Dion 2003] J. M. Dion. *Generic properties and control of linear structures systems: a survey*. Automatica, vol. 39, pages 1125–1144, 2003. 39
- [Djeziri 2007a] M. A. Djeziri. *Diagnostic des Systèmes Incertains par l'Approache Bond Graph*. PhD thesis, École Centrale de Lille, 2007. 48
- [Djeziri 2007b] M.A. Djeziri, R. Merzouki, B. Ould-Bouamama and G. Dauphin-Tanguy. *Robust Fault Diagnosis Using Bond Graph Approach*. Int. Journal of IEEE/ASME Transaction on Mechatronics, vol. 12 (6), pages 599–611, 2007. 111, 115, 132
- [Escobet 2001] T. Escobet and L. Travé-Massuyès. *Parameter estimation methods for fault detection and isolation*. In Proceeding of the 12th International Workshop on Principles of diagnosis, Via Lattea, Italy, 2001. 18
- [Feki 2008] M. El Feki, M. Di Loreto, E. Bideaux, D. Thomasset and W. Marquis-Favre. *On the role of essential orders on feedback decoupling and model inversion : bond graph approach*. In ECMS'08, Nicosia, Cyprus, June 2008. 38
- [Fotsu-Ngwompo 1997] Fotsu-Ngwompo. *Contribution au dimensionnement des systèmes sur des critères dynamiques et énergétiques - approche par Bond*

- Graph*. PhD thesis, Institut National des Sciences Appliquées de Lyon, 1997. 84, 85
- [Frank 1990] P. M. Frank. *Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results*. *Automatica*, vol. 26, no. 3, pages 459 – 474, 1990. xv, 5, 18
- [Frank 1997a] P. M. Frank and X. Ding. *Survey of robust residual generation and evaluation methods in observer-fault detection systems*. *Journal of Process Control*, vol. 7, no. 6, pages 403 – 424, 1997. xv, 5
- [Frank 1997b] P. M. Frank and B. K. Seliger. *Fuzzy logic and neural network applications to fault diagnosis*. *International Journal of Approximate Reasoning*, vol. 16, no. 1, pages 67 – 88, 1997. 16
- [Gao 1991] Z. Gao and P. J. Antsaklis. *On the Stability of the Pseudo-Inverse Method for Reconfigurable Control Systems*. *International Journal of Control*, vol. 53(3), pages 717–729, 1991. 22
- [Gawthrop 1995] P. J. Gawthrop. *Bicausal Bond Graphs*. In *International Conference on Bond Graph Modeling and Simulation (ICBGM'95)*, pages 83–88, January 1995. 45, 131
- [Gehin 2008] A. L. Gehin and M. Staroswiecki. *Reconfiguration Analysis Using Generic Component Models*. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 38, no. 3, pages 575 – 583, 2008. 29, 31, 32
- [Gertler 1997] J. Gertler. *Fault detection and Isolation Using Parity Relations*. *Control Eng. practice*, vol. 5, pages 653–661, 1997. xv, 5, 18
- [Hagenblad 2002] A. Hagenblad, F. Gustafsson and I. Klei. *A comparison of two methods for stochastic fault detection : The parity space approach and principal component analysis*. In *Proceeding of IFAC World Congress*, 2002. 18
- [Hallouzi 2008] R. Hallouzi. *Multiple-Model Based Diagnosis for Adaptive Fault-Tolerant Control*. PhD thesis, Technische Universiteit Delft, 2008. 5
- [Hsieh 2002] C. S. Hsieh. *Performance gain margins of the two-state LQ Reliable Control*. *Automatica*, vol. 38, pages 1985–1990, 2002. xv, 5, 12

BIBLIOGRAPHY

- [Hurdle 2009] E. E. Hurdle, L. M. Bartlett and J. D. Andrews. *Fault diagnostics of dynamic system operation using a fault tree based method*. Reliability Engineering and System Safety, vol. 94, pages 1371–1380, 2009. 18
- [InTraDE 2012] InTraDE. *Intelligent transportation for dynamic environment (InTraDE)*, <http://www.intrade-nwe.eu>, 2012, accessed 20 September 2012. 2, 10, 94
- [Isermann 1993] R. Isermann. *Fault diagnosis of machines via parameter estimation and knowledge processing Tutorial paper*. Automatica, vol. 29, no. 4, pages 815 – 835, 1993. 18
- [Isermann 1997] R. Isermann. *Supervision, fault-detection and fault-diagnosis methods - an introduction*. Control Eng. Practice, vol. 5, pages 639–652, 1997. 15
- [Isermann 2005] R. Isermann. *Model-based fault-detection and diagnosis-status and applications*. Annual reviews in Control, vol. 29, pages 71–85, 2005. 15
- [Isidori 1995] A. Isidori. Nonlinear control systems. Springer, Verlag Berlin Heidelberg, 1995. 29
- [Jardin 2008] A. Jardin, W. Marquis-Favre, D. Thomasset, F. Guillemard and F. Lorenz. *Study of a sizing methodology and a modelica code generator for the bond graph tool MS1*. In proceedings of the 6th International Modelica Conference, pages 125–134, 2008. 85
- [Jardin 2010] A. Jardin. *Contribution à une méthodologie de dimensionnement des Systèmes mécatroniques: analyse structurelle et couplage à l'optimisation dynamique*. PhD thesis, Institut National des Sciences Appliquées de Lyon, 2010. 84
- [Kam 2005] C. S. Kam and G. Dauphin-Tanguy. *Bond graph models of structured parameter uncertainties*. Journal of the Franklin Institute, vol. 342, pages 379–399, 2005. 132
- [Kanev 2004] S. Kanev. *Robust Fault-Tolerant Control*. PhD thesis, University of Twente, 2004. 13
- [Karnopp 1990] D. C. Karnopp, D. Margolis and R. Rosenberg. Systems dynamics: A unified approach. John Wiley, New York, second édition, 1990. 34

- [Kelly 2006] E. M. Kelly and L. M. Bartlett. *Application of the digraph method in system fault diagnostics*. In Proceedings of the First International Conference on Availability, Reliability and Security, 2006. 18
- [Khelassi 2009] A. Khelassi, D. Theilliol and P. Weber. *Reconfigurability Analysis for Reliable Fault-Tolerant Control Design*. In 7th Workshop on Advanced Control and Diagnosis, ACD'2009, Zielona Gora, Poland, 2009. xvi, 7, 28, 31, 32
- [Kwong 1995] W. A. Kwong, K. M. Passino, E. G. Laukonen and S. Yurkovich. *Expert Supervision of Fuzzy Learning Systems for Fault Tolerant Aircraft Control*. Proceeding of the IEEE, vol. 83, pages 466–483, 1995. xvi, 6
- [Larsson 1994] J. E. Larsson. *Hyperfast algorithms for model-based diagnosis*. In Computer-Aided Control System Design, 1994. Proceedings., IEEE/IFAC Joint Symposium on, pages 533–538, mar 1994. 18
- [Li 2000] B. Li, M. Y. Chow, Y. Tipsuwan and J. C. Hung. *Neural-network-based motor rolling bearing fault diagnosis*. Industrial Electronics, IEEE Transactions on, vol. 47, no. 5, pages 1060–1069, oct 2000. xv, 5, 16
- [Liao 2002] F. Liao, J. L. Wang and G. H. Yang. *Reliable Robust Flight Tracking Control: An LMI Approach*. IEEE Control Systems Magazine, vol. 10, no. 1, pages 76–89, 2002. xv, 5, 12
- [Lind 1990] M. Lind. *Representing Goals and Functions of Complex Systems - An Introduction to Multilevel Flow Modeling*. Rapport technique, Institute of Automatic Control System, Thechnical University of Denmark, Lyngby, 1990. 29
- [Loureiro 2012a] R. Loureiro, S. Benmoussa, Y. Touati, R. Merzouki and B. Ould-Bouamama. *Graphical Approach for State Reconstruction and Monitoring Analysis*. In The 7th IEEE Conference on Industrial Electronics and Applications, pages 1205–1210, Singapore, July 2012. 9
- [Loureiro 2012b] R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Bond graph model based on structural diagnosability and recoverability analysis: Application to intelligent autonomous vehicles*. IEEE Transactions on Vehicular Technology, vol. 61, no. 3, pages 986–997, 2012. 8, 30, 31, 38

BIBLIOGRAPHY

- [Loureiro 2012c] R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Extension of the Bond Graph Causality Inversion Method for Fault Detection and Isolation: Application to a Mechatronic System*. In Proceedings of the 8th IFAC International Symposium on Fault Detection, Supervision and Safety of Technical Processes, pages 150–155, Mexico city, Mexico, August 2012. 9
- [Loureiro 2012d] R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Structural Reconfigurability Analysis for an Over-Actuated Electric Vehicle*. In R. Merzouki, editeur, *Mechatronic & Innovative Applications* (www.eurekaselect.com/101969/volume/1). Bentham science, 2012. 8
- [Loureiro 2012e] R. Loureiro, R. Merzouki and B. Ould-Bouamama. *Structural Reconfiguration Conditions Based on Bond Graph Approach: Application to an Intelligent Autonomous Vehicle*. In Proceedings of the 8th IFAC International Symposium on Fault Detection, Supervision and Safety of Technical Processes, pages 970–975, Mexico city, Mexico, August 2012. 8
- [Maciejowski 2003] J. M. Maciejowski and C. N. Jones. *MPC Fault-Tolerant Flight Control Case Study: Flight 1862*. In Proceedings of the SAFE Process 2003: 5th Symposium on Fault Detection and Safety for Technical Processes, pages 121–126, 2003. 13, 23, 24
- [Maki 1997] Y. Maki and K. A. Loparo. *A neural-network approach to fault detection and diagnosis in industrial processes*. *Control Systems Technology, IEEE Transactions on*, vol. 5, no. 6, pages 529–541, 1997. 16
- [Maurya 2005] M. R. Maurya, R. Rengaswamy and V. Venkatasubramanian. *Fault diagnosis by qualitative trend analysis of the principal components*. *Chemical Engineering Research and Design*, vol. 83, pages 1122–1132, 2005. 17
- [Maurya 2007a] M. R. Maurya, R. Rengaswamy and V. Venkatasubramanian. *Fault diagnosis using dynamic trend analysis: A review and recent developments*. *Engineering Applications of artificial intelligence*, vol. 20, pages 133–146, 2007. 17
- [Maurya 2007b] M. R. Maurya, R. Rengaswamy and V. Venkatasubramanian. *A signed directed graph and qualitative trend analysis-based framework for incipient fault diagnosis*. *Chemical Engineering Research and Design*, vol. 85, no. 10, pages 1407–1422, 2007. 18

- [Medjaher 2005] K. Medjaher. *Contribution de l'outil bond graph pour la conception de systèmes de supervision de processus industriels*. PhD thesis, Université des Sciences et Technologie de Lille, 2005. 53
- [Mhaskar 2006] P. Mhaskar. *Robust Model Predictive Control Design for Fault-Tolerant Control of process Systems*. *Ind. Eng. Chem. Res.*, vol. 45, pages 8565–8574, 2006. xv, 5, 23, 24
- [Miksch 2008] T. Miksch, A. Gambier and E. Badreddin. *Real-time Performance Comparison of Fault-Tolerant Controllers*. In *IEEE Multi-conference on Systems and Control*, San Antonio, Texas (USA), September 2008. xv, 5, 13, 22, 23, 24
- [Moore 1981] B. C. Moore. *Principal component analysis in linear systems: controllability, observability and model reduction*. *IEEE Transactions on Automatic Control*, vol. 26, pages 17–32, 1981. 25
- [Moyes 1995] A. Moyes, G. M. Burt, J. R. McDonald, J. R. Capener, J. Dray and R. Goodfellow. *Application of expert systems to fault diagnosis IN ALTERNATORS*. In *Proc. Int. Conf. Elect. Mach. Drives*, 1995. 17
- [Niemann 2005] H. Niemann and J. Stoustrup. *Passive fault tolerant control of a double inverted pendulum - a case study*. *Control Engineering Practise*, vol. 13, pages 1047–1059, 2005. xv, 5, 12
- [Oca 2009] S. M. Oca, V. Puig, D. Theilliol and S. Tronil-Sin. *Fault-Tolerant Control Design using LPV Admissible Model Matching: Application to a two-degree of Freedom Helicopter*. In *17th Mediterranean Conference on Control and Automation*, Makedonia Palace, Thessaloniki, Greece, June 2009. 13
- [Ohman 1999] B. Ohman. *Failure model analysis using multilevel flow models*. In *Proceedings of the 5th European Control Conference*, Karlsruhe, Germany, 1999. 30
- [OKTAL 2012] OKTAL. *SCANeR Driving Simulation Engine*, <http://www.scanersimulation.com>, 2012, accessed 29 September 2012. 110
- [Ould-Bouamama 2003] B. Ould-Bouamama, A. K. Samantaray, M. Staroswiecki and G. Dauphin-Tanguy. *Derivation of Constraint Relations from Bond Graph Models for Fault Detection and Isolation*. In *International Conference on Bond Graph Modeling and Simulation*, pages 104–109, 2003. 2, 18, 35, 38, 48, 49

BIBLIOGRAPHY

- [Ould-Bouamama 2006] B. Ould-Bouamama, M. Staroswiecki and A.K. Samantaray. *Software for Supervision System Design In Process Engineering Industry*. In 6th IFAC IFAC Safeprocess, pages 691–695. IFAC, september 2006. 2
- [Pacejka 1991] H. B. Pacejka and R. S. Sharp. *Shear Force developments by Pneumatic tires in steady state conditions, A review of modelling aspects*. Vehicle systems dynamics, vol. 20, pages 121–176, 1991. 101
- [Padmakumar 2009] S. Padmakumar, Vivek Agarwal and Kallol Roy. *A Comparative Study into Observer based Fault Detection and Diagnosis in DC Motors: Part-I*. In Proceeding of World Academy of Science, Engineering and Technology, Rome, Italy, 2009. 18
- [Patton 1991] R. J. Patton and J. Chen. *A robust parity space approach to fault diagnosis based on optimal eigenstructure assignment*. In Proceeding International Conference Control, pages 1056–1061, 1991. 18
- [Patton 1997] R. J. Patton. *Fault-Tolerant Control Systems. The 1997 SITUATION*. In Proceedings of the 3rd IFAC symposium on fault detection, supervision and safety for technical processes, pages 1033–1055, August 1997. xv, 4, 12, 19
- [Paynter 1961] H. M. Paynter. *Analysis and design of engineering systems*. M.I.T. Press, 1961. 34
- [Qian 2003] Y. Qian, X. Li, Y. Jiang and Y. Wen. *An expert system for real-time fault diagnosis of complex chemical process*. Expert Systems with Applications, vol. 24, pages 425–432, 2003. 17
- [Qian 2008] Y. Qian, L. Xu, X. Li, L. Lin and A. Kraslawski. *LUBRES: An expert System development and implementation for real-time fault diagnosis of a lubricating oil refining process*. Expert Systems with Applications, vol. 35, pages 1252–1266, 2008. xv, 5, 17
- [R. Merzouki 2007] M.A. Djeziri R. Merzouki B. Ould-Bouamama and M. Bouteldja. *Modelling and Estimation of Tire-Road Longitudinal Impact Efforts using Bond Graph Approach*. Mechatronics, vol. 17, pages 93–108, 2007. 102
- [Rahmani 1993] A. Rahmani. *Etude structurel le des systmes linéaires par l’approche bond graph*. PhD thesis, Université des Sciences et Technologies de Lille, 1993. 39

- [Rahmani 1996] A. Rahmani, C. Sueur and G. Dauphin-Tanguy. *On the infinite structure of systems modelled by bond-graph: Feedback decoupling*. In IEEE SMC'96, Pekin, China, 1996. 81
- [Samantaray 2008a] A. K. Samantaray and S. K. Ghoshal. *Bicausal bond graphs for supervision: From fault detection and isolation to fault accommodation*. Franklin Institute, vol. 345, no. 1, pages 1–28, 2008. 38
- [Samantaray 2008b] A. K. Samantaray and B. Ould-Bouamama. Model-based process supervision. a bond graph approach. Springer Verlag, 2008. 7, 18, 34, 35, 39, 50, 53, 129, 131
- [Shaker 2011] H. R. Shaker and M. Tahavori. *Control reconfigurability of bilinear hydraulic drive systems*. In Fluid Power and Mechatronics (FPM), 2011 International Conference, pages 477–480, 2011. 25, 29, 31, 32
- [Soc 2001] Société de technologie Michelin. *The tyre Grip*, 2001. 101
- [Srinivas 1994] S. Srinivas. *A probabilistic approach to hierarchical model-based diagnosis*. In Proceedings UAI-94, pages 538–545, 1994. xv, 5, 16
- [Staroswiecki 1998] M. Staroswiecki and A. L. Gehin. *Analysis of system reconfigurability using generic component models*. In Control'98, Swansea (UK), 1998. 29
- [Staroswiecki 2000] M. Staroswiecki. *Quantitative and qualitative models for fault detection and isolation*. Mechanical Systems and Signal Processing, vol. 14, no. 3, pages 301 – 325, 2000. 18
- [Staroswiecki 2001] M. Staroswiecki and A. L. Gehin. *From control to supervision*. Annual Reviews in Control, vol. 25, pages 1–11, 2001. 40
- [Staroswiecki 2002] M. Staroswiecki. *On Reconfigurability with Respect to Actuator Failures*. In Proceeding of 15th Triennial World Congress, Barcelona, Spain, 2002. xvi, 7, 26, 27, 28, 31, 32
- [Staroswiecki 2008] M. Staroswiecki. *On fault handling in control systems*. International Journal of Control, Automation, and Systems, vol. 6, no. 3, pages 296–305, 2008. 39, 40

BIBLIOGRAPHY

- [Sueur 1991] C. Sueur and G. Dauphin-Tanguy. *Bond Graph Approach for Structural Analysis of MIMO Linear Systems*. Journal of the Franklin Institute, vol. 328, no. 1, pages 55–70, 1991. 2, 38, 48, 61
- [Tagina 1996] M. Tagina, J. P. Cassar, G. Dauphin-Tanguy and M. Staroswiecki. *Bond graph models for direct generation of formal fault detection systems*. System Analysis, Modelling and Simulation, vol. 23, no. 1-2, pages 1–17, June 1996. 48
- [Tharrault 2008] Y. Tharrault, G. Mourot and J. Ragot. *Fault detection and isolation with robust principal component analysis*. In Control and Automation, 2008 16th Mediterranean Conference on, pages 59 –64, june 2008. 16
- [Theilliol 2009] D. Theilliol, Y.M. Zhang and J.C. Ponsart. *Fault Tolerant Control System against Actuator Failures based on Re-configuring Reference Input*. In International Conference on Advances in Computational Tools for Engineering Applications, July 2009. 23
- [Thoma 2000] J. U. Thoma and B. Ould-Bouamama. Modelling and simulation in thermal and chemical engineering. bond graph approach. Springer Verlag, 2000. 34
- [Touati 2011] Y. Touati, B.O. Bouamama and R. Merzouki. *Robust residuals generation and evaluation using bond graph and linear filtering*. In Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on, pages 2318 – 2323, dec. 2011. 111
- [Touati 2012] Y. Touati, R. Merzouki and B. Ould-Bouamama. *Bond graph model based for fault estimation and isolation*. In Safeprocess 2012 (Accepted for oral presentation), 27 - 29 August 2012. 7, 64, 74, 79, 132, 138
- [Van-Dijk 1994] J. Van-Dijk. *On the Role of Bond Graph Causality in Modelling Mechatronic Systems*. PhD thesis, University of Twente, Den Haag, The Netherlands, 1994. 34
- [Venkatasubramanian 2003a] V. Venkatasubramanian, R. Rengaswamy, K. Yin and S. N. Kavuri. *A review of process fault detection and diagnosis Part I: Quantitative model-based methods*. Computers and Chemical Engineering, vol. 27, pages 293–311, 2003. 15

- [Venkatasubramanian 2003b] V. Venkatasubramanian, R. Rengaswamy, K. Yin and S. N. Kavuri. *A review of process fault detection and diagnosis Part II: Qualitative models and search strategies*. Computers and Chemical Engineering, vol. 27, pages 313–326, 2003. 15
- [Venkatasubramanian 2003c] V. Venkatasubramanian, R. Rengaswamy, K. Yin and S. N. Kavuri. *A review of process fault detection and diagnosis Part III: Process history based methods*. Computers and Chemical Engineering, vol. 27, pages 327–346, 2003. 15
- [Wang 2004] S. Wang and F. Xiao. *AHU sensor fault diagnosis using principal component analysis method*. Energy and Buildings, vol. 36, no. 2, pages 147 – 160, 2004. 16
- [Wu 1995] S. T. Wu and K. Youcef-Toumi. *On the relative degrees and zero dynamics from physical system modelling*. Journal of Dynamic Systems, Measurement and Control transactions of the American Society of Mechanical Engineering, vol. 117, no. 2, pages 205–217, 1995. 84
- [Wu 2000] N. E. Wu, K. Zhou and G. Salomon. *Control reconfigurability of linear time-invariant systems*. Automatica, vol. 36, pages 1767–1771, 2000. xv, xvi, 6, 7, 25, 28, 29, 31, 32
- [Xiaojun 2009] G. Xiaojun, Y. Shixi, Z. Yun and Q. Suxiang. *Development of multilevel flow model for power plant process fault diagnosis*. In Control and Decision Conference, pages 189–193, June 2009. 18
- [Yang 2006] Z. Yang. *Reconfigurability analysis for a class of linear hybrid systems*. In Proceedings of 6th IFAC Safeprocess, Beijing, PR China, 2006. 29, 31, 32
- [Yongli 2006] Z. Yongli, H. Limin and L. Jinling. *Bayesian networks-based approach for power systems fault diagnosis*. IEEE Transactions on Power Delivery, vol. 21, no. 2, pages 634 – 639, 2006. 16
- [Zhang 2001] Y. Zhang and J. Jiang. *Integrated Active Fault-Tolerant Control Using IMM Approach*. IEEE Transactions on Aerospace and Electronic Systems, vol. 37, pages 1221–1235, 2001. 22, 23

BIBLIOGRAPHY

- [Zhang 2002] Y. Zhang and J. Jiang. *An Active fault-tolerant control systems against partial actuator failures*. IEEE Proceedigs-Control Theory and Applications, vol. 149, no. 1, pages 95–104, 2002. 13, 22, 23
- [Zhang 2008] Y. Zhang and J. Jiang. *Bibliographical review on reconfigurable fault-tolerant control systems*. Annual Reviews in Control, vol. 32, pages 229–252, 2008. xiv, 4, 21
- [Zhang 2009] Z. Zhang and W. Chen. *Adaptive output feedback control of nonlinear systems with actuator failures*. Imformation Science, vol. 179, pages 4249–4260, 2009. xv, 5
- [Zhao 1997] Q. Zhao and J. Jiang. *Reliable tracking control systems design against actuator failures*. In Proceedings SICE Conference, Tokushima, 1997. 13

Bond Graph Model Based on Structural Diagnosability and Recoverability Analysis : Application to Intelligent Autonomous Vehicles

Abstract :

This work deals with structural fault recoverability analysis using the bond graph model. The objective is to exploit the structural and causal properties of the bond graph tool in order to perform both diagnosis and control analysis in the presence of faults. Indeed, the bond graph tool enables to verify the structural conditions of fault recoverability not only from a control perspective but also from a diagnosis one. In this way, the set of faults that can be recovered is obtained previous to industrial implementation. In addition, a novel way to estimate the fault by a disturbing power furnished to the system, enabled to extend the results of structural fault recoverability by performing a local adaptive compensation directly from the bond graph model. Finally, the obtained structural results are validated on a redundant intelligent autonomous vehicle.

Keywords: structural analysis, fault recoverability, fault diagnosis, intelligent systems, bond graph.

Résumé :

La présente thèse concerne l'étude structurelle pour le recouvrement du défaut par l'approche du bond graph. L'objectif est d'exploiter les propriétés structurelles et causales de l'outil bond graph, afin d'effectuer à la fois le diagnostic et l'analyse de la commande du système physique en présence du défaut. En effet, l'outil bond graph permet de vérifier les conditions structurelles de recouvrement de défauts pas seulement du point de vue de l'analyse de commande, mais aussi en considérant les informations issues de l'étape de diagnostic. Par conséquent, l'ensemble des défauts tolérés est obtenu en mode hors-ligne avant d'effectuer une implémentation réelle. En outre, en estimant le défaut comme une puissance perturbatrice fournie au système, ce qui permet d'étendre les résultats d'analyse structurelle pour le recouvrement du défaut à une compensation locale adaptative, directement à partir du modèle bond graph. Enfin, les résultats obtenus sont validés dans une application d'un véhicule autonome intelligent redondant.

Mots clés : analyse structurelle, recouvrement de défaut, diagnostic, systèmes intelligents, bond graph.
