FAULT TOLERANT SYSTEMS

Marcel Staroswiecki
Ecole Polytechnique Universitaire de Lille, University Lille I, France

Keywords: Fault detection and isolation, fault estimation, fault tolerance, fault accommodation, system reconfiguration, objective reconfiguration

Contents

Summary

This chapter introduces the main concepts and approaches used for the design of Fault Tolerant Systems. The impact of faults on the setting of control and estimation problems is first analyzed, and two fault tolerance strategies, namely fault accommodation and system reconfiguration are distinguished. According to the level of knowledge which is provided by the Fault Detection and Isolation algorithms about the faulty system, either an accommodation or a reconfiguration problem is set.

When no solution exists to these problems, it is necessary to change the system objective. Two control problems, namely model matching and optimal control are investigated, and the conditions and means by which they can receive fault tolerant solutions are detailed. Since structural properties like controllability and observability are very important for the design of control or estimation algorithms, it is of interest to analyze whether they remain true or are destroyed when faults occur. This leads to characterize fault tolerant system configurations either in a deterministic way (redundancy degree), or in a stochastic one (reliability, mean time to failure).

1. Introduction

In spite of many advances in the field of systems and control design, complex systems sometimes do not render the services they were designed for, or run out of control, creating situations whose effects range from energy and material waste to loss of production, damage to the environment or even loss of human lives.

Malfunctions are the result of many possible causes, which can be classified into design errors, implementation errors, human operator errors, wear, aging, environmental aggressions. Fault Detection and Isolation (FDI) is aimed at detecting such malfunctions in real time, as soon and as surely as possible, and at finding their root cause, by identifying the system component(s) whose operation mode is not nominal.

Fault Tolerance (FT) is concerned with the system behavior under fault situations. Namely, the *analysis* of fault tolerance answers the question whether a given system, in a given fault situation, is still able to achieve its objective(s), while the *design* of fault tolerance provides the system with the hardware architecture and the software mechanisms which allow, if possible, to achieve a given objective not only in normal operation, but also in given fault situations.

Therefore, fault tolerance is defined with reference to:

1) one (or several) system objective(s)

2) one (or several) given fault(s)

Traditional control engineering considers two kinds of objectives, associated with the control of the system and with the estimation of its variables in real time. Roughly speaking, each objective is associated with a system structural property, namely controllability (the ability of the system state - or a functional of the state - to be controlled by the inputs) and observability (the ability of the system state - or a functional of the state - to be estimated from the outputs). Accordingly, fault tolerance analysis / design is referred to as Fault Tolerant Control (FTC) when control objectives are of interest, and as Fault Tolerant Estimation (FTE) when estimation objectives are considered. However, FT with respect to any system objective may be considered, for example one could be interested in Fault Tolerant Monitoring (FTM), namely in the ability of the system to still perform Fault Detection and Isolation in the presence of a (certain set of) fault(s).

This chapter will mainly develop the FTC problem. FTE and FTM considerations are quite similar, since observability and controllability are dual notions. It is organized as follows. Section 2 presents the basic formulation of a control problem, and the way the *control* point of view has to be adapted when Fault Tolerant Control is addressed. Sections 3 and 4 address two control problems whose fault tolerance is analyzed, namely, the Model Matching problem (leading to the so-called pseudo-inverse method) and the Optimal Control problem, in the Linear Quadratic setting.

The nominal problems being addressed in details in *Classic Design Methods for Continuous LTI-Systems* and *Control of Linear Multivariable Systems*, nominal results are only shortly reminded, as a starting point for the introduction of fault tolerance issues. In Section 5, the impact of system reconfiguration on structural properties like observability and controllability is presented. Critical resources and the evaluation of Fault Tolerance by means of deterministic as well as probabilistic considerations are introduced. The example of a (linearized) CCV-type aircraft is given in Section 6, illustrating the Model Matching and the Optimal Control approaches. The Fault Tolerant Estimation problem is also illustrated, by means of the analysis of the observability property, when the system is reconfigured following sensor fault occurrences. Concluding remarks and directions for FTC research are given in Section 7.

2. Control and Fault Tolerant Control

Control algorithms implement the solution of control problems. Control problems can be set in many different formulations, according to the way the system objectives are expressed, and according to the constraints that the solution must satisfy.

Fault tolerant control implements the solution of control problems in which the system objectives are achieved, in spite of the occurrence of a pre-specified set of faults.

Therefore, a *Fault Tolerant Control (FTC) problem* is above all a *control problem.* In order to set properly the FTC problem, it is necessary to recall the way control problems are set.

2.1. Control Problem

2.1.1. Standard Control Problem

A standard control problem aims at finding a control law in a given set U , such that :

i) the controlled system will achieve one (or several) control objective(s) O ,

ii) while its behavior satisfies a set of constraints C .

Thus, the solution(s) of the problem is (are) completely defined by the triple $< O, C, U >$.

- The set of admissible control laws U defines the algorithms that can be implemented, e.g. open loop control (a mapping from the time domain to the control space), closed loop control (a mapping from the output × reference spaces to the control space), using continuous or discrete valued arguments for the variables, allowing for linear or non-linear, continuous or discontinuous, differentiable or non-differentiable mappings, etc.

- The objective(s) O define(s) what the system is expected to achieve, when controlled by the above mentioned control law. Expressing O may range from very general statements (e.g. achieve closed loop stability) to much more specific ones (e.g. reach a given point, on a given circular orbit around the earth, at a given time, for a space rendezvous).

- The constraints C are functional relations that the behavior of the controlled system must satisfy over time. They can be classified into equality and inequality constraints. The equations which describe the behavior of the system (e.g. its state equations) obviously form a set of equality constraints, for any control problem, since it is impossible to the controlled system not to satisfy them. They are expressed by algebraic and differential or difference equations, when continuous variables are considered, and by other models (automata, Petri nets), when discrete values are of interest. Inequality constraints express that some saturations act on the system admissible solutions (e.g. in the space rendezvous problem, there is a limitation on the energy consumed along any admissible trajectory).

2.1.2. The Control Problem with Uncertainties

In the standard control problem, the constraints are supposed to be perfectly known. However, this is not the case in many practical situations, because constraints depend on parameters and unknown inputs which may vary in time, which may be different from one system to another one, whose identification is subject to identification errors, etc.

Uncertainties are considered by introducing a parameter vector θ, and noting $C(\theta)$ the set of constraints whose parameters have the value θ.

When uncertainties are present, the actual parameters value θ_a may be different from θ_n, the nominal one (θ_a is unknown, but it is supposed that the set Θ to which it belongs is known). Obviously, there is no guarantee that the solution of the standard control problem $< O, C(\theta_n), U >$ will also solve $< O, C(\theta_a), U >$, and therefore the actual system may fail to achieve its objectives, even in the absence of faults.

Two approaches can be defined to deal with that problem. In the *robust control* approach, the problem is set as finding a control law in U, such that the objective O is achieved whatever the parameter $\theta \in \Theta$ in the constraints $C(\theta)$. Such a setting can be symbolized by the triple $< O, C(\Theta), U >$. In the *adaptive control* approach, one first estimates the "true" parameter $\hat{\theta} \in \Theta$, and then solves the problem $< O, C(\hat{\theta}), U >$.

2.2. Fault Tolerant Control Problem

Fault tolerant control is concerned with the control of the faulty system. This can be done by changing the control law without changing the system which is operated (fault adaptation, fault accommodation, controller reconfiguration are terms often encountered in the literature), or by changing both the control and the system (in this case system reconfiguration is used). Since the control algorithm just implements the solution of a given control problem for a given system, changing the control or the system means that *the control problem has been changed* as the result of faults.

In order to understand the different strategies that can be applied to the design of fault tolerant control, let us first consider the impact of faults on the control problem $< O, C(\theta), U >$ (extension to the triple $< O, C(\Theta), U >$ is straightforward) and then analyze the knowledge which is available to the control engineer about this impact. The two possible FTC strategies (namely fault accommodation and system reconfiguration) follow as a consequence of the available knowledge.

2.2.1. Impact of Faults

System objectives. The occurrence of faults does not change the system objectives. Indeed, the very nature of fault tolerant control is to still try to achieve these objectives, *in spite* of the faults in a given fault set.

1) when this is possible, the system is said to be fault tolerant, *with respect to these objectives and to these faults*. The control engineer's task is to design some control law which is able to do that.

2) when this is impossible, the system is not fault tolerant with respect to these objectives and these faults. However, it is not enough to stand with this conclusion. Indeed, the control engineer should provide, in this case, indications about what to do with the system, otherwise catastrophic behavior may take place. Since the current

objectives cannot be achieved, the problem is transformed into finding new objectives that are of interest in the current situation, and to design the control law which is able to achieve these new objectives. In other terms, when the system fails to be fault tolerant, *objective reconfiguration* has to be performed.

System constraints. The occurrence of faults will obviously change the constraints $C(\theta)$ of the control problem, either because the value of the parameters will change or because the constraints themselves will have a different structure. Therefore the control problem will be transformed from $< O, C_n(\theta_n), U >$ into $< O, C_f(\theta_f), U >$ where $C_n(\theta_n)$ is the set of nominal constraints and parameters, and $C_f(\theta_f)$ is the set of constraints and parameters associated with the faulty system.

Admissible control laws. The occurrence of faults may also change the set of admissible control laws for example when faults occur in the computing and communication devices in which they are implemented, or when they change the saturation level of actuators. Let U_f be the new set of admissible controls, while the nominal one is U_n.

2.2.2. Passive vs Active Fault Tolerant Control

Two major approaches can be distinguished in the design of FTC.

Passive fault tolerance. In *passive fault tolerance*, the control law is not changed when faults occur. Therefore, the ability of the system to achieve its given objective must be preserved, *using the same control law*, whatever the system situation (healthy or faulty). This means that the control law achieves the system objectives when the system is healthy (thus it solves $< O, C_n(\theta_n), U_n >$), as well as when the system is faulty (thus it also solves $< O, C_f(\theta_f), U_f >$). Implementing passive fault tolerance for a given set of faults means that there is a common solution to problem $< O, C_n(\theta_n), U_n >$ and to all problems $< O, C_f(\theta_f), U_f >$, $f \in F$, where F indexes the set of all the considered faults.

This is a very demanding condition, which can be fulfilled, in general, only for objectives associated with very low levels of performances (it is a so-called *conservative* approach). Note that since the control law is not changed, the passive fault tolerance approach is similar to the robust approach when uncertain systems are considered. Indeed, faults can be considered as uncertainties which affect the system parameters. The difference lies not only in the size and interpretation of these changes, but also in the fact that the very structure of the constraints may change as the result of faults. Therefore, it may be possible to find solutions to the robust control problem, because the sets of solutions of the control problems associated with the different values of the uncertain parameters do intersect, while this is not true in general for the fault tolerant control problem.

Active fault tolerance. In active approaches, *the system configuration and/or the*

control law is changed when faults occur, so that the ability of the system to achieve its given objective is preserved, *using a system configuration and/or a control law adapted to each fault situation*. In other words, each of the problems $< O, C_n(\theta_n), U_n >$ and $< O, C_f(\theta_f), U_f >$, $f \in F$, receives its own specific solution (when it exists), thus allowing for much more demanding objectives. However, for each of these problems to be solved, it has first to be properly set, i.e. the knowledge about $C_f(\theta_f)$ and U_f must be available.

This obviously calls for a Fault Detection and Isolation (FDI) layer. Detailed presentation of FDI approaches can be found in *Fault Diagnosis for Linear Systems, Fault Diagnosis for Nonlinear Systems, Design Methods for Robust Fault Diagnosis, Qualitative Methods in Fault Diagnosis, Statistical Methods for Change Detection*.and only the general features which are necessary for understanding the fault tolerant control problem statement are developed below. When $< O, C_f(\theta_f), U_f >$ has no solution, the objective cannot be achieved under the given fault situation, and objective reconfiguration has to be explored, as previously explained.

2.2.3. Available Knowledge

Providing information about the fault impact is the goal of the FDI algorithms. However, the power and efficiency of these algorithms is limited. *Fault detection* informs that the problem to solve is no longer $< O, C_n(\theta_n), U_n >$. *Fault isolation* informs about the subset of the constraints $C_n(\theta_n)$ which are unchanged (those associated with the still healthy components), and the subset $U_f \subseteq U_n$ of control laws which can still be used. The knowledge about the changed constraints calls for *fault estimation*, which is a new function to be considered. According to its performances, three cases must be considered:

a) the FDI algorithm provides an estimate $\hat{C}_f(\hat{\theta}_f)$, $\hat{U}_f$ of the fault impact. Then, the post-fault problem to be solved is the standard control problem $< O, \hat{C}_f(\hat{\theta}_f), \hat{U}_f >$. Note that, when a solution exists, there is still a risk that the actual faulty system (described by $C_f(\theta_f)$ and U_f) fails to satisfy the objectives O, although the available model of the faulty system does satisfy them.

b) the FDI algorithm provides an estimate $\hat{\Gamma}_f(\hat{\Theta}_f), U_f$ of the fault impact, where $\hat{\Gamma}_f$ is a set of possible constraints and $\hat{\Theta}_f$ is a set of associated parameters. Then the problem to be solved is the robust control problem $< O, \hat{\Gamma}_f(\hat{\Theta}_f), \hat{U}_f >$. When a solution exists, the actual faulty system will satisfy the objectives O provided the actual constraints $C_f(\theta_f) \in \hat{\Gamma}_f(\hat{\Theta}_f)$, otherwise, the same risk as above exists.

c) the FDI algorithm detects and isolates the faults, but it cannot provide any estimate of

the fault impact $C_f(\theta_f), U_f$. In that case, the control engineer is faced with the problem of designing the control of a partly unknown system, which is not possible.

Other possible and even worse cases are those where the FDI system detects the fault, but it cannot isolate it, nor is it able to provide any estimate, and the case where the FDI system does not even detect the fault. In the first case, the only possibility to keep mastering the system is to use objective reconfiguration, namely by moving to a fall back mode, while in the second case, any catastrophic behavior is possible. Active fault tolerance is only concerned with cases a), b) and c).

2.2.4. Active Fault Tolerant Control Strategies

Fault accommodation. Fault accommodation is the fault tolerant control strategy which is associated with cases a) and b). It solves the problem $< O, \hat{C}_f(\hat{\theta}_f), \hat{U}_f >$ or $< O, \hat{\Gamma}_f(\hat{\Theta}_f), \hat{U}_f >$, which are associated with the control of the faulty system. The fault situation can be accommodated with respect to the objectives O when the problem has a solution. Note that the interpretation of fault accommodation is that it is a strategy by which *the faulty system is controlled* in a specific way, so as to still achieve the objectives which were (before the fault) achieved by the healthy system. Solving the problem $< O, \hat{C}_f(\hat{\theta}_f), \hat{U}_f >$ or $< O, \hat{\Gamma}_f(\hat{\Theta}_f), \hat{U}_f >$ means that:

- the controlled system contains the same components as the healthy one, but some of them are faulty. More precisely, the faulty system has the same configuration as the healthy one, i.e. no component has been switched-off or switched-on in response to the fault(s).

- only the control law has been changed, since it now solves $< O, \hat{C}_f(\hat{\theta}_f), \hat{U}_f >$ or $< O, \hat{\Gamma}_f(\hat{\Theta}_f), \hat{U}_f >$ instead of $< O, C_n(\theta_n), U_f >$. This is why this strategy is also sometimes called control reconfiguration.

System reconfiguration. System reconfiguration is the fault tolerant control strategy which is associated with case c). Remember that in this case part of the faulty system is unknown. The only means to set a control problem is to switch-off the faulty components (which are known from the isolation function), and try to achieve the objectives using only the remaining (healthy) ones (among which some were possibly not in use in the pre-fault configuration). Let $C_f(\theta_f) = C_n'(\theta_n) \cup C_f''(\theta_f)$ where $C_n'(\theta_n)$ is the subset of the constraints which are associated with the healthy part of the system, and $C_f''(\theta_f)$ is the subset of the constraints which are associated with the faulty part. Note that $C_n'(\theta_n)$ are known while $C_f''(\theta_f)$ are unknown. Using similar notations, let $U_f = U_n' \cup U_f''$. Then, the reconfiguration strategy solves the problem $< O, C_n'(\theta_n), U_n' >$, i.e. it tries to achieve the system objectives by using only the healthy

part of the system.

Fault accommodation vs system reconfiguration. Fault accommodation can only be used when the FDI algorithm detects and isolates the faults, and estimates the model of the faulty system, while system reconfiguration only needs the fault detection and isolation functions. The choice of a reconfiguration strategy might follow from the impossibility of estimating the fault, or it can be deliberate, so as to implement fault tolerant strategies which provide guaranteed results (i.e. which work whatever the fault mode in the switched-off components), and are as simple and as understandable as possible by operators (indeed, controlling the faulty system might be possible, but would use ”strange” controls, that the operators would not understand, which might lead to dangerous reactions such as trying to counteract the accommodated control by manual interventions). It can also be noted that the analysis of system reconfiguration provides structural results, i.e. results which do not depend on the type of the fault, but only on the component(s) which is (are) faulty.

2.2.5. On-line vs Off-line Solution of the FTC Problem

It may be noticed that the reconfiguration strategy does not imply complex algorithms. Indeed, reconfiguration is (by definition) the choice of a new (post-fault) configuration (and an associated control law) which is able to carry out the system objective instead of the (pre-fault) one. The control of the post-fault configuration may of course be computed on-line, but since there is only a finite number of such possible configurations, it may also be computed off-line and a bank of controllers implemented, such that the appropriate control is switched on line when the associated fault is recognized.

When fault accommodation is considered, the new control has to be computed on-line and the control problem associated with the specific faulty system has to be solved under real time constraints. In some cases, possible faults are known in advance from the Fault Mode and Effect Analysis (FMEA), and the accommodated control can be computed off-line and applied on-line with the actual situation’s specific parameters, as soon as the model associated with the faulty system becomes available from the FDIE algorithm.

In all cases, very important issues are associated with post-fault stability, since the faulty system will be controlled by the nominal control as long as the accommodation or reconfiguration procedure has not produced a solution.

2.3. Supervision Problem

Suppose that both the accommodation and the reconfiguration strategies fail to provide a solution. In this case, another objective has to be provided to the system. *Objective reconfiguration* introduces the most general problem, defined by the triple, $< O, C(\Theta), U >$ where O is a set of possible control objectives. This is a supervision problem, i.e. a fault tolerant control problem associated with a decision problem: when faults are such that fault tolerance cannot be achieved, the system goal itself has to be changed.

There are many ways by which new objectives can be defined. The simplest one is associated with purely quantitative changes, which only affect the values of some parameters, like e.g. in the controllability problem : when transferring the system state from $x(t_0) = x_0$ to $x(t_f) = x_f$ is no longer possible because, as the result of faults, x_f does no longer belong to the controllable subspace, an objective reconfiguration might be to replace x_f by the nearest (in some sense) state x_f^* which belongs to the still controllable subspace.

More drastic objective reconfiguration are of course possible, e.g. in the problem of filling a tank in minimum time for processing a batch in food industry. Assume that the fraction of the nominal maximum flow that the pump is still able to deliver, $u'_{\max}$, is so low that the batch would be oxidized before the treatment can start, then obviously the objective of keeping the system in production can no longer be achieved, and has to be replaced by the objective of restoring the nominal system capability (which is a maintenance objective).

As it can be seen from such examples, in most cases, human operators are involved in the definition of new system objectives. It may happen that no achievable objective exists under the actual system possibilities. This can be a design error, or a deliberate choice to accept certain failure scenarios, e.g. for reasons of benefit or small likelihood of certain events. Note that fail-to-safe conditions are intended to avoid this case, since they express that for certain classes of faults, the objective of stopping the system must always be achievable.

3. Model Matching and the Pseudo-inverse Method

In the following sections, model matching and optimal control objectives will be considered for linear systems described by the constraints

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{1}$$

where $x(t) \in R^n$ and $u(t) \in R^m$.

In model matching, the control objective O is to design a control law such that the closed loop system behavior follows the reference model

$$\dot{x}(t) = Mx(t) + Ne(t), \tag{2}$$

where the pair M, N is given, and $e(t)$ is an arbitrary input signal.

When state feedback is considered, the set of admissible controls U is defined by

$$u(t) = Ge(t) - Kx(t), \tag{3}$$

where K, G are to be defined.

-
-
-

Bibliography

[1] Jiang, B., M. Staroswiecki and V. Cocquempot (2004). Fault Diagnosis Based on Adaptive Observer for a Class of Nonlinear Systems with Unknown Parameters. *Int. J. of Control*, 77 (4):415-426. [This paper describes an observer-based approach to fault estimation in nonlinear uncertain systems. Fault estimation is a prerequisite to fault accommodation].

[2] Blanke M., M. Kinnaert, J. Lunze and M. Staroswiecki (2003). *Diagnosis and Fault Tolerant Control.* Springer-Verlag, ISBN 3-540-01056-4. [One of the first books embracing the whole field of fault detection associated with fault tolerant control].

[3] Gao Z. and P.J. Antsaklis (1991). Stability of the pseudo-inverse method for reconfigurable control systems. *Int. J. of Control*, 53:717-729. [This paper describes an improvement of the pseudo-inverse method for insuring the stability of the fault accommodated system].

[4] Gehin A. - L. and M. Staroswiecki (1999). A formal approach to reconfigurability analysis. Application to the three tank benchmark. In Proc. of European Control Conference, ECC'99, Karlsruhe. [This paper describes a generic component based model, in which services are provided by components under several versions. For given faults, the system is reconfigurable if and only if the services that are necessary for the objective to be achieved can still be run (the resources of at least one of their versions are still available)].

[5] Gehin A.-L., M. Assas and M. Staroswiecki (2000). Structural analysis of system reconfigurability. In Proc. of IFAC Symp. on Fault Detection Supervision and Safety of Technical Processes, Safeprocess'2000, 292-297. [This paper proposes a structural analysis approach for determining if the system is still controllable / observable after faults have occurred].

[6] Huang R. F. and C. Y Strangel (1990). Restructurable control using proportional-integral implicit model following. *J. Guidance, Control and Dynamics*, 13:303-309. [One of the first papers on controller reconfiguration by model-matching].

[7] Maciejowski J. M. (1997). Reconfigurable Control Using Constrained Optimization, In Proc. of European Control Conference ECC'97, Brussels, Belgium, Plenary Lectures and Mini-Courses, 107 – 130. [This paper presents the way by which the model-predictive control approach can encompass fault tolerance issues, by changing the constraints that the solution has to satisfy in response to the actual situation of the system].

[8] Mahmoud M., J. Jiang and Y. Zhang (2003). *Active Fault Tolerant Control Systems. Stochastic Analysis and Synthesis.* Springer-Verlag, Lecture Notes in Control and Information Sciences, ISBN 3-540-00318-5. [This book analyses fault tolerant control systems, from the point of view of their stochastic properties which follow from the fact that faults are random events].

[9] Niemann H. H. and J. Stoustrup (2002). Reliable control using the primary and dual Youla parameterization. In Proc. of 41st IEEE Conf. Decision and Control, 4353-4358. [In this paper, the architecture of fault tolerant control is based on the primary Youla parameterization of all stabilizing compensators, and the dual Youla parameterization is used to quantify the performance of the fault tolerant system].

[10] Patton R. J. (1997). Fault Tolerant Control : the 1997 situation. In Proc. of IFAC Symp. on Fault Detection Supervision and Safety of Technical Processes, Safeprocess'97, 1033-1055. [This plenary

lecture gives a state of the art of Fault Tolerant Control in 1997].

[11] Rauch H. E. (1995). Autonomous Control Reconfiguration. *IEEE Control Systems Magazine*, 15 (6) : 37-48. [this paper gives a survey of fault accommodation approaches].

[12] Staroswiecki M. , G. Hoblos and A. Aitouche (2004). Sensor Network Design for Fault Tolerant Estimation. *Int. J. of Adaptive Control and Signal Processing*, 18:55-72. [This paper presents the fault tolerant estimation problem, and an algorithm for the design of sensor networks with specified fault tolerance properties].

[13] Wu N. E., Y. M. Zhang and K. Zhou (2000). Detection, estimation and accommodation of loss of control effectiveness. *Int. J. of Adaptive Control and Signal Processing*, 14(7):775-795. [This paper presents an integration of FDIE and FTC for fault accommodation of a special case of actuator faults].

[14] Wu N. E. (2004). Coverage in fault-tolerant control. *Automatica*, 40:537-548. [This paper analyzes the events that occur between the onset of a subsystem failure and the settlement of the accommodated / reconfigured control. It evaluates the probability that the system will successfully recover after a fault.]

[15] Zhang Y. M. and J. Jiang (2001). Integrated active fault tolerant control using IMM approach. IEEE Tr. on Aerospace and Electronic Systems, 37(4):1221-1235. [This paper presents an approach in which pre-computed control laws are selected, based on an estimation of the system impairment status].

[16] Zhang Y. M. and J. Jiang (2002). An active fault tolerant control system against partial actuator failures. IEE Proc. - Control Theory and Applications, 149(1):95-104. [This paper presents an approach to FTC based on eigenstructure assignment].

Biographical Sketch

Marcel Staroswiecki was born in Melitopol (Ukraina) in 1945. He obtained the Engineering Degree from the Ecole Nationale Supérieure d'Ingénieurs des Arts et Métiers (with a silver medal distinction), in 1968. He then obtained a PhD in Automatic Control in 1970, and the French « Doctor es Sciences » degree in Physical Sciences in 1979.

He joined the University of Lille I as an Assistant Professor in 1969 and he became full Professor in 1983. He currently teaches at the University's Engineering School (Ecole Universitaire des Ingénieurs de Lille), and he is the Director of the Laboratoire d'Automatique et d'Informatique Industrielle de Lille (a shared laboratory between Ecole Centrale de Lille, Université des Sciences et Technologies de Lille and CNRS).

His research interests are on Fault Detection, Isolation and Recovery (FDIR) algorithms for complex systems. He heads a research group in which both the model based and the pattern recognition based approaches are developed. A special attention is paid to the implementation of FDIR procedures, especially within the frame of intelligent instruments and components. Prof. Staroswiecki is a member of two IFAC Technical Committees: Safeprocess and Intelligent Components and Instruments.

His activity has also been intensively dedicated to technology transfer, especially in direction of SME's. He created and was the director of the Lille University Technology Transfer Center in the field of Automatic Control from 1982 to 1994, which performed more than 100 joint research contracts between Lille University and industrial companies. For this activity, he was distinguished by the « Grand Prix Edmond Faucheur de la Société des Industriels du Nord de la France ».

He was a « chargé de mission » at the French Ministry of research in the field of Automatic Control and Robotics from December 1993 to December 1996. There he initiated different policies for the development of the research and of the doctoral activity in French laboratories.

Prof. Staroswiecki obtained the French distinction of « Chevalier de l'Ordre des Palmes Académiques » in 1990 and became « Officier dans l'ordre des Palmes Académiques » in 1995.