



NONLINEAR MODEL DECOMPOSITION FOR ROBUST FAULT DETECTION AND ISOLATION USING ALGEBRAIC TOOLS

Denis Berdjag, Cyrille Christophe, Vincent Cocquempot, Bin Jiang

► To cite this version:

Denis Berdjag, Cyrille Christophe, Vincent Cocquempot, Bin Jiang. NONLINEAR MODEL DECOMPOSITION FOR ROBUST FAULT DETECTION AND ISOLATION USING ALGEBRAIC TOOLS. International Journal of Innovative Computing, Information and Control, 2006, 2 (6), pp.1337-1354. hal-00198435

HAL Id: hal-00198435

<https://hal.science/hal-00198435>

Submitted on 17 Dec 2007

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

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

NONLINEAR MODEL DECOMPOSITION FOR ROBUST FAULT DETECTION AND ISOLATION USING ALGEBRAIC TOOLS

D. BERDJAG¹, C. CHRISTOPHE¹, V. COCQUEMPOT¹ AND B. JIANG²

¹ LAGIS, Polytech'Lille

Université des Sciences et Technologies de Lille

Cité Scientifique, 59655 Villeneuve d'Ascq cedex, France

denis.berdjag@ed.univ-lille1.fr, {christophe.cyrille,vincent.cocquempot}@univ-lille1.fr

² College of Automation Engineering

Nanjing University of Aeronautics and Astronautics

29 YuDao Street, 210016 Nanjing, P.R. China

binjiang@nuaa.edu.cn

ABSTRACT. This paper is concerned with fault detection and isolation in nonlinear dynamic systems. A structural decomposition of the system is made, using the criterion of robustness to unknown inputs. To achieve this decomposition, special mathematical tools are used. The algebraic nature of these tools permits a more general formulation of the residual generation problem. These mathematical tools are presented with illustrations and the unknown input-free decomposition method is described. Some aspects of the method are discussed, and solutions are proposed to improve the existing algorithm. An application on an academic example is also provided.

1. Introduction. Process monitoring is a major concern for the industrial world. The early detection and localization of faulty elements can help preventing larger failures or even the destruction of the monitored plant, by stopping the control of the process or by using an adapted control law. This is the main task of fault detection and isolation (FDI) in dynamic systems. Because of the great relevance of this thematic in industrial plants, FDI has become a fundamental issue of research for the automatic community during the last two decades.

Model-based FDI rely on analytical redundancy obtained from the mathematical model of the system to monitor. This redundancy is used to generate fault indicators named residuals. These signals allow to check discrepancies between the behavior of the system and its mathematical model. Among the well-established concepts to generate residuals are parameter estimation approach [4, 5], parity space approach [2, 3] and observer-based approach [1, 11] with the latter being the most relevant. All of these approaches use traditionally a linear model of the plant, even if the behavior of the system is nonlinear. The main reason of that issue is the difficulties encountered when dealing with nonlinear equations. That's why in the last decade, a consequent amount of researches were initiated to deal with those problems. Two major directions were investigated : The most common are geometric methods [6, 7] where differential geometric tools are used. The alternative direction is represented by algebraic methods [17, 9]. While geometric tools have proved their efficiency , their application is restricted by a certain class of nonlinearities. From this perspective, algebraic methods provide more general solutions. In

particular, a set of algebraic tools was developed in [12] to handle discrete model decomposition with respect to some criteria. In [9], these tools were extended to the continuous-time case and to the FDI problem. These mathematical tools, referred as *algebra of functions*, permits to express function relationship in algebraic terms. The author uses these tools to solve the problem of unknown input-free decomposition. However, some aspects of the former method remain unclear, in particular the computational aspect.

The objective of this paper is to investigate the *Fundamental Problem of Residual Generation* (FPRG) using an *algebra of functions* approach. A parallel is made with the classical solution of this problem [1, 3] which relies on design of residual generators bank. An approach based on model decomposition is proposed instead. This approach permits to split the solution in two steps : robust decomposition of the model and stability of the residual generator. This sequential method makes the solution of the FPRG problem more flexible. Also, some computation aspects of the decomposition method from [9] are detailed and extended. To summarize, the contributions of this work are :

- The Fundamental Problem of Residual Generation is solved using model decomposition with a robustness / sensitivity criterion.
- Computation enhancements of *algebra of functions* operators are proposed.
- The decomposition algorithm from [9] is revisited. The initialization step is improved in order to lead to a complete solution.
- An nonlinear example to illustrate initialization problem is provided and a complete solution is given.

The paper is organized as follows : In section 2, the robust residual generation problem is expressed. In section 3, the algebra of functions is presented. The decomposition with a robustness/sensitivity criterion is investigated in section 4. Section 5 addresses the contributions of this work : the computational aspect as well as an improvement of the decomposition method. In section 6, an application on an academic example is provided to show the effectiveness of the proposed computation methods. Conclusions and perspectives on future works close the paper.

2. Problem Formulation. Consider the following system

$$\Sigma : \begin{cases} \dot{x}(t) = f(x(t), u(t), \gamma(t), \rho(t)) \\ y(t) = h(x(t)) \end{cases} \quad (1)$$

where $x \in X \subseteq \mathbb{R}^n$ is the state, $u \in U \subseteq \mathbb{R}^m$ is the input and $y \in Y \subseteq \mathbb{R}^l$ is the output. The fault signals to detect are represented by the vector ρ while the other faults will be considered as perturbations and stacked along with unknown inputs (UI) into the unknown vector γ . In order to achieve fault detection, a residual $r(t)$ is generated. This signal is related to the fault ρ . It is equal to zero in the fault-free case and different from zero if the fault ρ occurs.

$$\begin{cases} \rho(t) = 0 & \Rightarrow & r(t) = 0 \\ \rho(t) \neq 0 & \Rightarrow & r(t) \neq 0 \end{cases} \quad \forall \gamma, \forall x(t), y(t), u(t) \quad (2)$$

However, in real-life systems, the residual will not exactly equal zero for the fault-free case. This issue is caused by noise and modelization errors. To deal with this case, decision methods must be used to state if a fault is occurring or not when the residual is not exactly null. This

topic is discussed in [8].

In the multiple fault case, fault detection and isolation is traditionally achieved by designing a residual generators bank using the complete model [18]. Each residual is sensitive to specific faults and robust to the unknown inputs and to all the other faults. It means that each residual generator corresponds to a particular configuration of the vectors ρ and γ . Multiple configurations are obtained by permutation of vector components of ρ and γ .

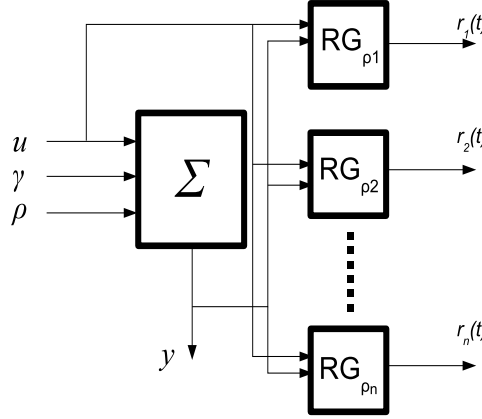


FIGURE 1. Robust residual generator bank

This traditional structure is represented in figure (1). Every residual $r_i(t)$ must fulfill the following conditions

- Robustness to the perturbation vector γ_i
- Sensitivity to the fault vector ρ_i
- Asymptotic convergence in the fault free case. if $\rho_i = 0$, $\forall \gamma_i$ $\lim_{t \rightarrow \infty} r_i(t) = 0$

where ρ_i , γ_i are respectively the fault vector and the perturbation vector corresponding to the fault i . This set of conditions is known as *the Fundamental Problem of Residual Generation* (FPRG).

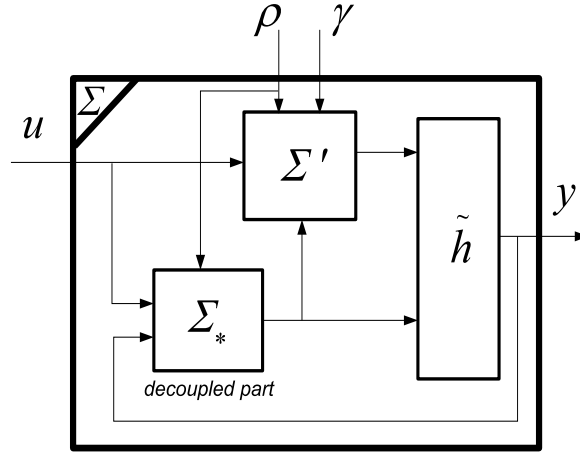
It must be noticed that the FPRG constraints are applied in one single step in the traditional RG design method. This usually leads to solve a multiple constraint problem which is often challenging.

As an alternative, to reduce design complexity, we propose to decompose the design process in two steps. First of all, the decomposition of the complete model with respect to the robustness/sensitivity constraint is performed. The model Σ is decomposed in two parallel dynamics Σ_* and Σ' as shown in figure (2)

The subsystem Σ_* is the part of Σ which is robust to the perturbation vector γ and sensitive to ρ .

$$\Sigma_* : \begin{cases} \dot{x}_*(t) = f_*(x_*(t), y(t), u(t), \rho(t)) \\ y_*(t) = h_*(x_*(t)) \end{cases} \quad (3)$$

The decoupling is performed using an output injection which must be determined. In order to keep a link with the real system y , the outputs y_* must be expressed as a combination measured

FIGURE 2. Parallel decomposition of Σ

outputs y .

$$y_*(t) = h_*(x_*(t)) = c(y(t)) \quad (4)$$

The system Σ' represents the complementary subsystem of Σ_* . To determine the decoupled subsystem Σ_* a decomposition function is applied on Σ .

Let ϕ be a differentiable function that is defined as follows

$$\begin{aligned} \phi &: X \rightarrow X_* \\ &: x(t) \mapsto x_*(t) \end{aligned} \quad (5)$$

where X and X_* are respectively the state spaces of Σ and Σ_* . The function ϕ determines the subsystem Σ_* upon Σ

$$\begin{aligned} \Sigma &\xrightarrow{\phi} \Sigma_* \\ \phi(x(t)) &= x_*(t) \end{aligned} \quad (6)$$

To achieve fault detection, a residual generator is designed upon the robust subsystem Σ_* . The corresponding residual will inherit naturally the robustness property.

$$\Sigma_0: \begin{cases} \dot{\hat{x}}_*(t) &= f_0(\hat{x}_*(t), \hat{y}_*(t), y(t), u(t)) \\ r(t) &= c(y(t)) - h_*(\hat{x}_*(t)) \end{cases} \quad (7)$$

where c is the function from (4). The function f_0 is obtained from f_* 3 using a feedback. The determination of the feedback function is a classical problem of observer design [19] and is not be addressed here.

In the case of multiple faults, a bank of robust subsystems is designed (*see fig. (3)*). Each subsystem is robust to a given subset of faults and sensitive to remaining faults.

In comparison with the methodology illustrated figure (1), the residual generators are easier to design due to the two steps synthesis. At the same time, the decomposition process is simpler because of reduced constraints.

In this section, we have shown that the FPRG problem can be expressed as a problem of decomposition with respect to a robustness/sensitivity criterion. This is not a new result and

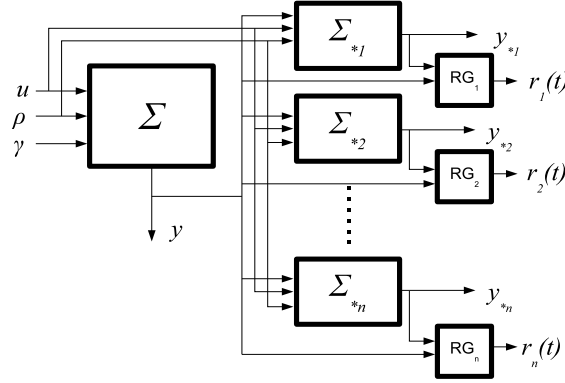


FIGURE 3. Robust subsystems bank + RG

it is similar to the formulations suggested in [6, 9]. However, a slight difference exist : the proposed formulation emphasis the independent role of the robust subsystem instead focusing on the subsystem as a part of a residual generator.

Consider the following relation

$$\frac{d\phi}{dx} f(x, u, \gamma, \rho) = f_*(\phi(x), h(x), u, \rho) \quad (8)$$

obtained from relation (6) using derivation and substitution of $\dot{x}_*$, $\dot{x}$ with their expressions in (1) and (3). Equation (8) expresses mathematically the problem of robust decomposition, and serves as a beginning point to many Residual generators bank-based approaches [6, 16]. However, authors use geometric tools to determine the transformation, and consequently are limited to a class of nonlinear systems. In order to solve (8) in the general case, a set of algebraic tools is introduced : the *algebra of functions* (AF).

3. Algebra of Functions. The algebra of functions is a set of mathematical tools that can express function relationship. These tools were developed initially by Zhirabok and Shumsky [9, 10] as an extension to the continuous-time and infinite sets cases of the pair algebra [12] used to describe the behavior of sequential automata. The principal feature of the algebra of functions is versatility. It permits to keep the same notations and methods to process different types of models : linear and nonlinear, continuous-time and discrete-time.

Algebra of functions is a set of operators and operations applied on vector functions defined on a subset $X \subseteq \mathbb{R}^n$. The set of all functions on X is denoted D_X . Three types of operators and operations are involved: ordering, interaction and advanced ones. These operators and operations are defined in the following and a geometrical illustration is given for clearness sake.

3.1. Ordering : Relations $\leq$ and $\cong$. Consider the functions $\alpha : X \rightarrow S$ and $\beta : X \rightarrow T$ with $S, T \subseteq X$. We denote $\alpha \leq \beta$ if and only if there is a function $v : S \rightarrow T$ such as $v(\alpha) = v \circ \alpha = \beta$.

$$\alpha \leq \beta \wedge \beta \leq \alpha, \Rightarrow \alpha \cong \beta \quad (9)$$

Remark 3.1. The elements of D_X are ordered by the relation $\leq$. The smallest element of D_X is the element generating all the remaining elements. On the opposite, the greatest element is the element which can be generated by any other element of D_X .

Illustration

Let $\alpha : X \longrightarrow S$ and $\beta : X \longrightarrow T$ be two differentiable vector functions. Assume that

$$\begin{aligned} A &= \text{span}\{d\alpha_1, \dots, d\alpha_k\} \\ B &= \text{span}\{d\beta_1, \dots, d\beta_l\} \end{aligned} \quad (10)$$

where α_i, β_i are the components of α and β respectively. The functions α and β correspond geometrically to the co-distributions A and B . In a similar manner, the following statements correspond

$$\begin{aligned} A \supseteq B &\longleftrightarrow \alpha \leq \beta \\ A = B &\longleftrightarrow \alpha \cong \beta \end{aligned} \quad (11)$$

3.2. Interaction : Operations $\times$ and $\square$. These operations define respectively the greatest (in the sense of the operation $\leq$) function generating α and β and the smallest function which can be generated by α and β simultaneously.

Let $\Omega = \{f | f \leq \alpha, f \leq \beta\}$ and $\Psi = \{l | \alpha \leq l, \beta \leq l\}$. We denote

$$\alpha \times \beta = g \iff g \in \Omega \wedge \forall f \in \Omega : f \leq g \quad (12)$$

$$\alpha \square \beta = s \iff s \in \Psi \wedge \forall l \in \Psi : s \leq l \quad (13)$$

The main properties of these operators are

- $\alpha \leq \beta \iff \alpha \times \beta \cong \alpha \iff \alpha \square \beta \cong \beta$
- if $\alpha \leq \delta$ and $\beta \leq \delta$ then $\alpha \square \beta \leq \delta$

Illustration

Let $\alpha : X \longrightarrow S$ and $\beta : X \longrightarrow T$ be two differentiable vector functions. A and B are the co-distributions spanned respectively by $d\alpha$ and $d\beta$. The following statements correspond

$$\begin{aligned} A \cap B &\longleftrightarrow \alpha \times \beta \\ A \cup B &\longleftrightarrow \alpha \square \beta \end{aligned} \quad (14)$$

Figure (4) illustrates the presented concepts

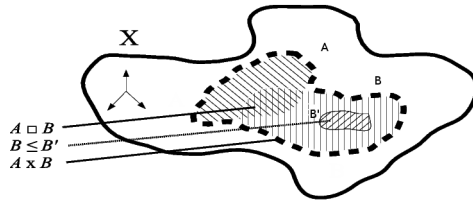


FIGURE 4. Geometric illustration

3.3. Advanced : Pair relation Δ , operators m and M . Consider the differentiable function β and some function g . A function α forms a pair with β with respect to g i.e. $(\alpha, \beta) \in \Delta_f$ if and only if there is a function $v : S \rightarrow T$ such as

$$v(\alpha(x), u) = \frac{d\beta}{dx} f(x, u) \quad (15)$$

with $x \in X$ and $u \in U$

$m(\alpha)(x)$ is the smallest function (in the sense of the operation $\leq$) forming a pair on the right with α , i.e.

$$(\alpha, m(\alpha)) \in \Delta_f \text{ and } \forall (\alpha, \beta) \in \Delta_f : m(\alpha) \leq \beta \quad (16)$$

By analogy, $M(\beta)$ is the greatest function forming a pair on the left with β

$$(M(\beta), \beta) \in \Delta_f \text{ and } \forall (\alpha, \beta) \in \Delta_f : \alpha \leq M(\beta) \quad (17)$$

The main properties of these two operators are

- if $\alpha \leq \beta$ then $m(\alpha) \leq m(\beta) \Leftrightarrow$
if $\alpha \cong \beta$ then $m(\alpha) \cong m(\beta)$
- $M(\alpha \times \beta) \cong M(\alpha) \times M(\beta)$
- $\alpha \leq M(m(\alpha)), m(M(\beta)) \leq \beta$

Remark 3.2. *The operator $m(\alpha)$ represents the maximum amount of available information on the next state of the system, knowing a certain amount of information in the actual state α . The operator $M(\beta)$ represents the minimum amount of information necessary to express the next system state β . These operators are useful to determine the behavior of a subsystem inside the system. For example, if some state vector components are influenced by an unknown input, it is possible to determine if these components are necessary or not to determine the next state of a perturbation-free subsystem, and to guarantee its decoupling properties.*

Illustration Let $\alpha : X \rightarrow S$ and $\beta : X \rightarrow T$ be two differentiable vector functions. A and B are the co-distributions spanned respectively by $d\alpha$ and $d\beta$. Consider the following system

$$\begin{cases} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{cases} \quad (18)$$

where $x \in X$, $u \in U$ and f, g are some functions. Some specificities of the system (18) can be expressed using differential geometric operators like Lie derivative. Notations concerning Lie algebra and differential geometric operators can be found in [13]. Exact correspondance between the Algebra of function operators and Lie algebra operators is difficult to show in a short paragraph, nevertheless a parallel can be made on how the characteristics of the system (18) are expressed using the two sets of operators.

- A is invariant (involutive) under the dynamics $f \Leftrightarrow L_f(A) \subset A \iff (\alpha, \alpha) \in \Delta_f$
- A is (h, f) invariant $\iff (\alpha \times h, \alpha) \in \Delta_f$
- $L_f(\alpha_i) \iff M(\alpha_i)$
where α_i are the row components of α

It must be noticed that the application of the presented algebraic operators isn't restricted to systems such as (18), as in the case with differential geometric operators.

4. Decomposition with robustness/sensitivity criterion. The decomposition problem with robustness/sensitivity criterion problem is expressed using *algebra of functions*. The provided solution is sequential and is based on the determination of a robust and invariant subsystem.

An algebraic reformulation of the decomposition criteria is made. These condition are used to design a sequential decomposition method.

Consider the relation (8). This equation expresses mathematically the decoupling problem. The decomposition problem will be solved if a function ϕ that verifies (8). However, taken alone this condition alone do not guarantee the robustness of the resulting subsystem.

4.1. Decomposition conditions. Invariance

The equation 3 is rewritten using the operation $\times$

$$\begin{aligned} f_*(x_*, y, u, \rho) &= f_*(\phi(x), h(x), u, \rho) \\ &= f_*((\phi \times h)(x), u, \rho) \end{aligned} \quad (19)$$

Equation (8) becomes

$$f_*((\phi \times h)(x), u, \rho) = \frac{d\phi}{dx} f(x, u, \gamma, \rho) \quad (20)$$

This relation corresponds to a pair (15) : $(\phi \times h, \phi) \in \Delta_f$. From here, using the definition (17), the following relation can be written

$$\phi \times h \leq M(\phi) \quad (21)$$

This relation expresses the invariance condition in *algebra of function*. The output function h means that the decoupling is obtained by an output injection *i.e.* that f_* is (h, f_*) invariant (involutive). We will see later how to implement this condition in the decomposition.

4.2. Robustness. The subsystem Σ_* is robust to the unknown input vector γ

$$\frac{\partial}{\partial \gamma} f_*(x_*, y, u, \rho) = 0 \Rightarrow \frac{\partial}{\partial \gamma} \frac{d\phi}{dx} f(x, u, \gamma, \rho) = 0 \quad (22)$$

Let ϕ^0 a function from D_X that verifies the condition (22)

$$\phi^0 \leq \phi \Leftrightarrow \phi \in D_X \wedge \frac{d\phi}{dx} \frac{\partial f}{\partial \gamma} = 0 \quad (23)$$

The function ϕ^0 is the greatest function dimension-wise that achieves the decoupling. Every single function that verifies (22) is necessarily a part of ϕ^0 .

$$\phi^0 \leq \phi \quad (24)$$

Remark 4.1. It must be noticed that the subsystem obtained using ϕ^0 is structurally free from the unknown input vector γ but by no means it means that this system is feasible or designable. To obtain a designable subsystem, the function ϕ^0 must verify the invariance condition.

4.3. Sensitivity. The decoupled system Σ_* is sensitive to the fault vector ρ

$$\frac{\partial}{\partial \rho} f_*(x_*, y, u, \rho) \neq 0 \Rightarrow \frac{\partial}{\partial \rho} \frac{d\phi}{dx} f(x, u, \gamma, \rho) \neq 0 \quad (25)$$

Let β^0 a function from D_X that *do not* verify the condition (25)

$$\beta^0 \leq \beta \Leftrightarrow \beta \in D_X \wedge \frac{d\beta}{dx} \frac{\partial f}{\partial \rho} = 0 \quad (26)$$

The subsystem Σ_* will be sensitive to faults if the decomposition function ϕ is not entirely part of β^0

$$\beta^0 \not\leq \phi \quad (27)$$

or completely independent from β^0

$$\beta^0 \square \phi = \text{const} \quad (28)$$

4.4. Residual generation capability. When performing the decomposition of the system, if $\dim(Y)/\text{leqdim}(X)$, the link with the measured outputs y may disappear. If it is the case, residual generation is impossible. To avoid this particular case, the subsystem output y_* must be a combination of the measured outputs.

$$y_* = c(h(x)) = h_*(\phi(x)) \Rightarrow \begin{cases} \phi \leq c \circ h \\ \phi \square h \neq \text{const} \end{cases} \quad (29)$$

for some function c .

The residual will be robust to the perturbation if $\phi^0 \leq \phi$. The transformation ϕ must be generated by ϕ^0 . This basic function ϕ^0 must be determined. It will also need to be invariant. The following theorem, introduced in [14], permits to determine the invariant part of ϕ^0

Theorem 4.1. *Consider the following recurrence*

$$\phi^{i+1} = m(\phi^i \times h) \square \phi^i$$

with $i = 0, 1, \dots, n$. When $\phi^{k+1} \cong \phi^k$ for some step k , then $\phi^* = \phi^k$ is the smallest function fulfilling

$$\begin{cases} \phi^0 \leq \phi^* \\ m(\phi^* \times h) \leq \phi^* \end{cases} \quad (30)$$

Using this theorem, a function ϕ^* is obtained. This function is robust and invariant. It is the first candidate to achieve the decomposition. In some particular cases, one can use ϕ^* to obtain the decoupled system.

4.5. The decomposition method. The set of decomposition conditions can be summarized as follows

Decomposition conditions

$$\begin{cases} \phi^* \leq \phi \\ \phi \square h \neq \text{const} \\ \phi \square \beta^0 = \text{const} \\ \phi \times h \leq M(\phi) \end{cases} \quad (31)$$

Using these condition, an algorithm to build of the transformation ϕ can be designed. It must be noticed that the decomposition condition is the stopping rule and must be checked at every step ; if sensitivity is not verified, the subsystem Σ_* is robust to γ and ρ

Decomposition methodology

Initialization

The first step is to determine the greatest (dimension-wise) decoupling function ϕ^0 . In the literature [9], ϕ^0 is directly determined from (22) and is always linear. This point will be discussed in the next section. The sensitivity condition must be checked $\beta^0 \not\leq \phi^0$ in order to continue the decomposition

Finding the invariant and robust part

The following theorem, introduced in [14], computes the largest (dimension-wise) invariant part of ϕ^0

Theorem 4.2. *Consider the following recurrence*

$$\phi^{i+1} = m(\phi^i \times h) \square \phi^i$$

with $i = 0, 1, \dots, n$. When $\phi^{k+1} \cong \phi^k$ for some step k , then $\phi^ = \phi^k$ is the smallest function fulfilling*

$$\begin{cases} \phi^0 \leq \phi^* \\ m(\phi^* \times h) \leq \phi^* \end{cases} \quad (32)$$

Using this theorem, a function ϕ^* is obtained. This function is robust and invariant. It is the first candidate to achieve the decomposition. In some particular cases, one can use ϕ^* to obtain the decoupled system.

Designing the robust subsystem

In some cases, the robust and invariant decomposition function ϕ^* may lead to an infeasible subsystem. To determine the largest part of ϕ^* which guarantees a feasible subsystem Σ_* the following algorithm is used

Algorithm 1

BEGIN

1. $\phi_1 = c^0 \circ h, j = 1$
2. if $h \times \phi_1 \times \dots \times \phi_j \leq M_f(\phi_j)$ then $k = j$, go to step 4

3. Find the greatest ϕ_{j+1} such as $\phi^* \leq \phi_{j+1}$ and $h \times \phi_1 \times \dots \times \phi_{j+1} \leq M_f(\phi_j)$
 inc(j)
 go to step 2
4. $\phi = \phi_1 \times \phi_2 \times \dots \times \phi_k$

END

Row functions $\phi_1, \phi_2, \dots, \phi_3$ are the components of ϕ . The function c^0 is the largest function (dimension-wise) verifying (29). The resulting function ϕ will give us the largest feasible robust subsystem Σ_* . It must be noticed that the use of another function c replacing c^0 will result in a different decomposition function.

5. Implementation problems. In the previous sections, the different steps of the unknown input-free decomposition were given. The purpose of the previous publications on that topic [10, 9] were leading clearly to a theoretical description of algebraic decomposition methodology. However, many points remain unclear especially the implementation. In particular, the computation of the different operators needs to be detailed. Also, an important step of the decomposition algorithm - the initialization - is not sufficiently discussed. This paper's contribution is the clarification of these two particular points.

5.1. Computation of the operators of Algebra of functions. In the following, methods to calculate *algebra of function* operators are provided. Some ideas are taken from [10] and an enhanced method to compute the operator m is proposed.

Computing the operation $\times$: Consider the following independent functions $\alpha : X \rightarrow S$ and $\beta : X \rightarrow T$

$$\alpha \times \beta = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (33)$$

Computing the operation $\square$: Consider the functions $\alpha : X \rightarrow S$ and $\beta : X \rightarrow T$. Let a function g exist such as $g = \alpha \square \beta$. It means the functional components of g ; namely g_i will satisfy the following equalities

$$\begin{aligned} \text{rank} \begin{bmatrix} \frac{d\alpha}{dx} \\ \frac{dg_i}{dx} \end{bmatrix} &= \text{rank} \begin{bmatrix} d\alpha \\ dx \end{bmatrix} \\ \text{rank} \begin{bmatrix} \frac{d\beta}{dx} \\ \frac{dg_i}{dx} \end{bmatrix} &= \text{rank} \begin{bmatrix} d\beta \\ dx \end{bmatrix} \end{aligned} \quad (34)$$

where $\frac{d}{dx}$ are the Jacobian matrices. These equalities gives partial differential equations concerning each component g_i of the function g i.e.

$$g = g_1 \times \dots \times g_i \quad (35)$$

Computing the operator M : Let remind the formal definition of the operator M

$$(M(\beta), \beta) \in \Delta_f \text{ and } \forall (\alpha, \beta) \in \Delta_f : \beta \leq M(\beta) \quad (36)$$

for some vector function f . In the context of this paper, the function f is the state function of the system to process.

Assume the following statement

$$\frac{d\beta}{dx} f(x, u) = \Psi(a_1(x), a_2(x), \dots, a_p(x), u) \quad (37)$$

If $\Psi = \sum a_i(x) b_i(u)$ and all the b_i are linearly independent then

$$M(\beta) = a_1 \times a_2 \times \dots \times a_p \quad (38)$$

If the function Ψ has a more general form, an additional condition must be fulfilled to use the relation (38) : the existence of control values $c_1, c_2, \dots, c_p$ such as

$$\frac{d\beta}{dx} f(x, c_1) \times \dots \times \frac{d\beta}{dx} f(x, c_p) \leq a_i \quad (39)$$

with $i = 1..p$.

Computing the operator m : Let remind the formal definition of the operator m

$$(\alpha, m(\alpha)) \in \Delta_f \text{ and } \forall (\alpha, \beta) \in \Delta_f : m(\alpha) \leq \beta \quad (40)$$

To calculate the operator m , the partial differential equations deduced from the following equality must be solved

$$\text{rank} \left[\frac{d\alpha}{dx} \right] = \text{rank} \left[\frac{dm_i(\alpha)}{dx} \frac{df}{dx} + f^T \left(\frac{d}{dx} \left(\frac{dm_i(\alpha)}{dx} \right) \right)^T \right] \quad (41)$$

$m_i(\alpha)$ is the corresponding row component of the function m_i . The equality (41) is immediately deduced from

$$v(\alpha(x), u) \leq \frac{dm(\alpha)(x)}{dx} f(x, u) \quad (42)$$

for some vector function v .

This type of differential equations is not obvious to solve for the general case. The assumption of

$$\frac{d}{dx} \left(\frac{dm_i(\alpha)(x)}{dx} \right) = 0 \quad (43)$$

can be made to obtain a linear solution if it exists, and will lead to the following equalities

$$\frac{dm_i(\alpha)}{dx} \frac{df}{dx} = N \frac{d\alpha}{dx} \quad (44)$$

for some matrix N .

Remark 5.1. *The computations are far from simple if the function α has a general form. In the context of this paper, $\alpha = \phi^0 \times h$. And ϕ^0 is determined using the relation (23). Disregarding the dimension problems discussed in the fifth section, this form do not simplify the computations as well.*

However, we can add the knowledge of the system we had to simplify the computations. If the model of the system is polynomial, these particular polynomials can be used to compute the operator m by using the following property

$$\alpha \cong \beta \Rightarrow m(\alpha) \cong m(\beta) \quad (45)$$

The basic idea is to substitute the function α which gives a complicate result when computing $m(\alpha)$, with the function β which gives a simpler solution. The substitution function must be chosen with consideration to the state function $f(x, u)$. If $f(x, u)$ has the following form

$$f_i(x, u) = a_{(i,1)}(x)b_{(i,1)}(u) + \dots + a_{(i,w)}(x)b_{(i,w)}(u) \quad (46)$$

where $f_i(x, u)$ are the row components of $f(x, u)$, then the function β is obtained as follows

$$\beta = a_{(1,1)}(x) \times \dots \times a_{(i,v)}(x) \quad (47)$$

with $a_{(1,1)}(x), \dots, a_{(i,v)}(x)$ being independent and the relation $\alpha \leq a_{(i,j)}(x)$ being fulfilled for $i = 1..n$ and $j = 1..v$.

Example 5.1. A simple illustration of function substitution is provided. The function $f(x, u)$ is the same as in (57).

$$\begin{aligned} \alpha = \begin{bmatrix} x_1 \\ x_4 \\ x_5 \end{bmatrix} &\cong \beta = \begin{bmatrix} x_4^2 \\ x_4 x_5 \\ x_1 x_4 x_5 \end{bmatrix} \Rightarrow m \left(\begin{bmatrix} x_1 \\ x_4 \\ x_5 \end{bmatrix} \right) \cong m \left(\begin{bmatrix} x_4^2 \\ x_4 x_5 \\ x_1 x_4 x_5 \end{bmatrix} \right) \\ m(\alpha) &= \begin{bmatrix} \frac{x_2 x_3}{x_1} \\ \frac{1}{x_5} \\ \frac{x_1 x_5}{x_4} \end{bmatrix}, m(\beta) = \begin{bmatrix} x_1 - x_2 \\ x_1 - x_2 - x_3 \\ x_4 \end{bmatrix} \end{aligned} \quad (48)$$

5.2. An enhanced initialization of the decomposition algorithm. While algebraic tools give us a proper formalization to the decomposition problem, the definition of the set of transformation candidates ϕ^0 remains inadequate. If no candidate function exists to fulfill the robustness condition (23), it will not necessarily imply that the robust decomposition is impossible. The model must be rewritten in an adequate form to allow the decomposition. If the decomposition remains impossible even with the transformation, we can say that the decomposition using algebra of function is really impossible.

The first step is to check the *transformability* of the system into adequate form. We need to know if the influence of the unwanted signals could be reduced. The following relation is checked

$$\text{rank} \left[\frac{\partial f(x, u, \rho, \gamma)}{\partial \gamma} \right] = n \quad (49)$$

If the condition (49) is verified, then the system is already in adequate form, so the straightforward use of the decomposition algorithm is possible. Otherwise, a transformation is required. This transformation aims the elimination of the perturbation from a maximal number of states. The variables to eliminate are the components of perturbation vector γ . The system is transformed as follows

$$f(x, u, \rho, \gamma) \longrightarrow \tilde{f} : \begin{cases} \tilde{f}_1(\tilde{x}, u, \rho, \gamma) \\ \tilde{f}_2(\tilde{x}, u, \rho) \end{cases} \quad (50)$$

Several cases are considered, according to the type of unwanted signals injection, linear or nonlinear. In this paper, the linear injection case is processed. In the general case, nonlinear elimination techniques [15] are applied to reduce the influence of perturbation, using Groebner bases algorithm for example. This topic is one of the future works of the authors.

Consider the following system for the linear case

$$\Sigma : \begin{cases} \dot{x} = f(x, u, \rho) + E\gamma \\ y = h(x) \end{cases} \quad (51)$$

where E is a linear matrix. According to condition (22), the initial set of transformation candidates ϕ^0 will be larger if the influence of γ on the state equation is reduced. Consider now a linear matrix T which transforms the fault injection matrix E

$$TE = \dot{E} = \begin{bmatrix} D \\ 0 \end{bmatrix} \quad (52)$$

where D is a diagonal matrix. The system (51) is transformed in consequence

$$f(x, u) + K\rho + E\gamma \xRightarrow{T} \tilde{\Sigma} : \begin{cases} \tilde{f}_1(\tilde{x}, u, \rho) + D\gamma \\ \tilde{f}_2(\tilde{x}, u, \rho) \end{cases} \quad (53)$$

Considering that $\text{rank}[E] = \text{rank}[D] = r$, the transformation T will generate $n - r$ states that are free from the influence of the unwanted signal γ . This transformation must be determined by using variable elimination techniques on the matrix E . If E is linear, Gaussian elimination is performed.

The determination of the matrix T is not always obvious. When performing Gaussian elimination algorithm on the matrix E , the matrix $\dot{E}$ is obtained. In the general case, the matrix E is rectangular so the straightforward

$$T = \dot{E}E^{-1} \quad (54)$$

cannot be used. The solution in this case is the use of the pseudo-inverse of E

$$E^+ = (E^T E)^{-1} E^T \quad (55)$$

under the condition that E is a full column rank matrix. The matrix T is given by

$$T = \dot{E}E^+ \quad (56)$$

The purpose of the transformation T is to give us a system which improves the determination of the set of candidate functions ϕ^0

6. An illustrative example. To illustrate the method, an academic example is processed. The system Σ to monitor is influenced by two faults and no perturbations. A *sign* function appears in the first state equation restricting the use of geometric approaches to decomposition. Consider the following system :

$$\begin{cases} \dot{x}_1 = x_2x_3 + \text{sign}(x_1)u_1 + \rho_2 \\ \dot{x}_2 = x_4^2 + \rho_1 + \rho_2 \\ \dot{x}_3 = x_4x_5 + \rho_2 \\ \dot{x}_4 = x_1x_4x_5 + x_2x_3u_2 \\ \dot{x}_5 = x_4 + x_3u_3 \end{cases}, \begin{cases} y_1 = x_1 \\ y_2 = x_4 \\ y_3 = x_5 \end{cases} \quad (57)$$

Where ρ_1 and ρ_2 represent the faults. To achieve detection and isolation of ρ_1 and ρ_2 a bank of two subsystems Σ_{*1} and Σ_{*2} is created. The subsystems Σ_{*1} , Σ_{*2} are robust respectively to ρ_2 and ρ_1 . The determination of Σ_{*1} is detailed below and Σ_{*2} is given.

The determination of Σ_{*1}

The robust function ϕ^0 is determined using the relation (23) with $\gamma = \rho_2$

$$\phi^0 = [x_4|x_5]^T \quad (58)$$

$\phi^0(x)$ represents the states decoupled from the fault ρ_2 . By analogy, the states free from the fault ρ_1 form $\beta^0(x)$ (26)

$$\beta^0 = [x_1|x_3|x_4|x_5]^T \quad (59)$$

The determination of the invariant part of ϕ^0 ; namely the function ϕ^* , is achieved by the application of the theorem 1.

The first iteration gives

$$\begin{aligned} \phi^1 &= \phi^0 \square m(\phi^0 \times h) \\ &= [x_4|x_5]^T \square m([x_1|x_4|x_5]^T) \\ &= [x_4]^T \end{aligned} \quad (60)$$

Since $\phi^1 \not\cong \phi^0$, a second iteration will be initiated

$$\begin{aligned} \phi^2 &= \phi^1 \square m(\phi^1 \times h) \\ &= [x_4] \square m([x_1|x_4|x_5]^T) \\ &= [x_4] \end{aligned} \quad (61)$$

Since $\phi^1 \cong \phi^2$, $\phi^* = [x_4]$. Unfortunately, it is clear that $\beta^0 \leq \phi^*$. It means that the decoupled system will be insensitive to the fault ρ_1 .

The solution lies in the transformation of the state equations discussed in section 5. The objective is to minimize the influence of $\gamma = \rho_2$ without transforming the outputs. In this particular case, Gaussian elimination is performed on the states x_1 , x_2 and x_3 . Using the method described in the previous section on the initial system, the following matrix T is obtained

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (62)$$

Consider now the system transformed using $T : \tilde{x} = Tx$

$$\begin{cases} \dot{\tilde{x}}_1 = \tilde{x}_2\tilde{x}_3 + \text{sign}(\tilde{x}_1)u_1 + \rho_2 \\ \dot{\tilde{x}}_2 = \tilde{x}_2\tilde{x}_3 - \tilde{x}_4^2 + \text{sign}(\tilde{x}_1)u_1 - \rho_1 \\ \dot{\tilde{x}}_3 = \tilde{x}_4^2 - \tilde{x}_4\tilde{x}_5 + \rho_1 \\ \dot{\tilde{x}}_4 = \tilde{x}_1\tilde{x}_4\tilde{x}_5 + \tilde{x}_2\tilde{x}_3u_2 \\ \dot{\tilde{x}}_5 = \tilde{x}_4 + \tilde{x}_3u_3 \\ \tilde{y}_1 = \tilde{x}_1 \\ \tilde{y}_2 = \tilde{x}_4 \\ \tilde{y}_3 = \tilde{x}_5 \end{cases} \quad (63)$$

The decomposition method is applied on (63). The new functions ϕ^0 and $\beta^0(x)$ are determined

$$\phi^0 = [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4|\tilde{x}_5]^T \quad (64)$$

$$\beta^0 = [\tilde{x}_1|\tilde{x}_4|\tilde{x}_5]^T \quad (65)$$

$\beta^0 \square \phi^0 \neq \text{const}$, so a decoupled system may exist.

The new function ϕ^* is determined using the theorem 1

The first iteration gives

$$\begin{aligned} \phi^1 &= \phi^0 \square m(\phi^0 \times h) \\ &= [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4|\tilde{x}_5]^T \square m([\tilde{x}_1|\tilde{x}_2|\tilde{x}_3|\tilde{x}_4|\tilde{x}_5]^T) \\ &= [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4]^T \end{aligned} \quad (66)$$

Since $\phi^1 \not\cong \phi^0$, a second iteration will be initiated

$$\begin{aligned} \phi^2 &= \phi^1 \square m(\phi^1 \times h) \\ &= [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4]^T \square m([\tilde{x}_1|\tilde{x}_2|\tilde{x}_3|\tilde{x}_4|\tilde{x}_5]^T) \\ &= [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4]^T \end{aligned} \quad (67)$$

Since $\phi^1 \cong \phi^2$, $\phi^* = [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4]^T$. It is clear that $\beta_0 \not\cong \phi^*$, so the decoupled system can be sensitive to the fault.

The next step is to apply the decomposition algorithm to obtain the decoupled system Σ_*

BEGIN

1. $\phi_1 = \phi^* \square h = [\tilde{x}_4]$
2. $h \times \phi_1 = [\tilde{x}_1|\tilde{x}_4|\tilde{x}_5]^T \not\leq M(\phi_1)$
with $M(\phi_1) = [\tilde{x}_1\tilde{x}_4\tilde{x}_5|\tilde{x}_2\tilde{x}_3|\tilde{x}_1\tilde{x}_3|\tilde{x}_1\tilde{x}_2|\tilde{x}_1^2|\tilde{x}_2^2]^T$

3. Let $\phi_2 = [\tilde{x}_2|\tilde{x}_3]^T$. It is clear that $\phi_2 \leq \phi^*$ and $h \times \phi_1 \times \phi_2 \leq M(\phi_1)$
4. $h \times \phi_1 \times \phi_2 = [\tilde{x}_1|\tilde{x}_2|\tilde{x}_3|\tilde{x}_4|\tilde{x}_5]^T \leq M(\phi_2)$
 with $M(\phi_2) = [\tilde{x}_2\tilde{x}_3|\tilde{x}_1\tilde{x}_3|\tilde{x}_1\tilde{x}_2|\tilde{x}_1^2|\tilde{x}_2^2|\tilde{x}_4\tilde{x}_5|sign(\tilde{x}_1)]^T$.
 The condition is satisfied and ϕ is given by
5. $\phi = \phi_1 \times \phi_2 = [\tilde{x}_2|\tilde{x}_3|\tilde{x}_4]^T$.

END

Since $\phi \square \beta^0 = \tilde{x}_4 \neq const \Leftrightarrow \beta^0 \not\leq \phi$, the subsystem Σ_{*1} is sensitive to the fault ρ_1 . The description of Σ_{*1} robust to ρ_2 and sensitive to ρ_1 is given by :

$$\begin{cases} \dot{x}_{*11} = x_{*11}x_{*12} - y_2^2 + sign(y_1)u_1 - \rho_1 \\ \dot{x}_{*12} = y_2(y_2 - y_3) + \rho_1 \\ \dot{x}_{*13} = y_1y_2y_3 + x_{*11}x_{*12}u_2 \end{cases} \quad (68)$$

with $y_{*11} = x_{*3}$ and $c_1(y) = y_2$

The system (69) is robust to the fault γ and sensitive to the fault ρ . To synthesize a residual generator a feedback is added to the system Σ_{*1} (see fig. 2). The feedback gain matrix can found using well known methods [19]. The obtained residual generator is structurally robust.

The subsystem Σ_{*2} robust to ρ_1 and sensitive to ρ_2 is determined in the same manner :

$$\begin{cases} \dot{x}_{*21} = y_2y_3 + \rho_2 \\ \dot{x}_{*22} = y_2 + x_{*21}u_3 \end{cases} \quad (69)$$

with $y_{*21} = x_{*22}$ and $c_2(y) = y_3$

This is an alternative to the classical synthesis of robust residual generator based on the complete model 57 which is more difficult to design from a computational perspective.

7. Conclusion. Classical residual generator design approaches are bound to the type of model to monitor. The discussed algebraic approach is not limited by this point and may be used in the same manner on different types of models. Its principal feature appears : versatility. A general formulation of the model decomposition with robustness/sensitivity criterion is possible which leads as shown in section 2 to a solution of the FPRG problematic. This is especially useful for nonlinear systems where no such general formulations exist.

Differential geometry-based methods are usually employed in this particular case. However, geometric methods are limited to a certain class of nonlinear systems. Also, the computation of the transformation is not clearly given. This transformation is delimited by a set of geometrical constraints. A particular solution is chosen, usually in an empirical way, to fulfill the constraints. A lot of research has been done on ways to eliminate or to accommodate this problem [7, 11, 16, 17, 18], and the present paper is aiming to the same direction.

An adapted formulation of the FPRG problem is proposed. Based on this formulation, the problem of model decomposition with robustness/sensitivity criterion is presented. The algebraic approach presented in [9] is discussed and extended. In particular, the implementation

aspects are detailed and the decomposition method is enhanced, especially the linear perturbation injection case. Our future works are summarized in the following points:

- The nonlinear perturbation injection case
- Extending the application of the algebra of functions by processing other problems : linearization using input/output injection with robustness criterion and decomposition of systems with mixed dynamics (continuous/discrete).
- The use of the robust subsystems is not bound by (FPRG) problematic. These subsystems may be employed for control purposes. Applications on Fault Tolerant Control (FTC) may be considered also.

REFERENCES

- [1] Patton, R., Robust model-based fault diagnosis : The state of the art., *Proceedings IFAC Symposium SAFE-PROCESS'94*, Vol. 1 , pp. 1-24 , 1994.
- [2] Gertler J., Analytical Redundancy Methods in Fault Detection and Isolation, Survey and Synthesis in *IFAC Safeprocess91*, vol. 1, p. 9-21, Baden Baden, Germany, 1991.
- [3] Staroswiecki, M. and Comtet-Varga, G., Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems, *Automatica* , Vol. 37 , pp. 687-699 , 2001.
- [4] Isermann, R., Process Fault Detection Based on Modeling and Estimation Methods, in *Automatica*, 20(4), pp. 387-404, 1984.
- [5] Isermann, R., Fault Diagnosis of Machines via Parameter Estimation and knowledge processing, in *Automatica*, 29(4), pp. 815-835, 1993.
- [6] DePersis, C. and Isidori, A., A geometric approach to nonlinear fault detection and isolation *IEEE Transactions on Automatic Control*, Vol. 46 , pp. 853-865 , 2001.
- [7] Jiang, B., Staroswiecki, M. and Cocquempot, Fault Diagnosis Based on Adaptive Observer for a Class of Nonlinear Systems with Unknown Parameters, *International Journal of Control*, Vol. 77 , pp. 415-426 , 2004.
- [8] Zhang, Q., Basseville, M. and Benveniste, A., Fault Detection and Isolation in Nonlinear Dynamical Systems: a Combined Input-Output and Local Approach, in *Automatica*, 34(11), pp. 1359-1374, 1998.
- [9] Zhirabok, A., Fault detection and isolation: linear and nonlinear systems, *Proceedings of the IFAC Symposium Safeprocess'97*, pp. 903-908, Hull, 1997.
- [10] Zhirabok, A. and Shumsky, A. Ye., A new mathematical techniques for nonlinear systems research, *Proceedings 12th World Congress IFAC*, pp. 485-488, Sidney, 1993.
- [11] Hammouri, H., Kinnaert, M. and El Yaagoubi, E., Application of nonlinear observers to fault detection and isolation. in *New Directions in Nonlinear Observer Design*, Springer-Verlag, pp. 423-443, 1999.
- [12] Hartmanis, J. and Stearns, R., *The algebraic structure theory of sequential machines*, Prentice-Hall , New York, 1966.
- [13] Isidori, A., *Nonlinear control theory. An introduction* , Springer-Verlag, Berlin, 1989.
- [14] Shumsky, A. Ye., Fault isolation in nonlinear dynamic systems by functional diagnosis, *Automation and Remote Control* , Vol. 12, pp. 148-155 , 1991.
- [15] Cox, D., Little, J. and O'Shea, *Ideals, Varieties, and Algorithms* , D. Springer Verlag , 1991.
- [16] Join, C.; Ponsart, J. and Sauter, D., Nonlinear filters bank for FDI: an alternative to the fundamental problem of residual generation, in Proc. 5th IFAC Symposium SAFEPROCESS'03, pp. 645-650, Washington D.C., 2003.
- [17] Fliess, M. and Join, C., An algebraic approach to fault diagnosis for linear systems, *Proc. CESA Conf.*, 2003.
- [18] Frank, P., Shreier, G. and Garcia, E.A., Non-linear observers for fault detection and isolation in *New directions in nonlinear observer design*, Springer-Verlag , pp. 399-422 , 1999.
- [19] Misawa, E. A. and Hedrick, J. K., Nonlinear observers - A state-of-the-art survey *Proc. of the Symposium, ASME Winter Annual Meeting*, pp. 59-69., Chicago, IL; USA , 1988.