Aalborg Universitet

AALBORG UNIVERSITY
D E N M A R K

Fault Detection and Isolation Using Analytical Redundancy Relations for the Ship Propulsion Benchmark

Izadi-Zamanabadi, Roozbeh

Publication date:
1998

Document Version
Også kaldet Forlagets PDF

Link to publication from Aalborg University

Citation for published version (APA):
Izadi-Zamanabadi, R. (1998). *Fault Detection and Isolation Using Analytical Redundancy Relations for the Ship Propulsion Benchmark*. Department of Control Engineering.

Fault Detection and Isolation Using Analytical Redundancy Relations for the Ship Propulsion Benchmark

An internal Report

by

Roozbeh Izadi-Zamanabadi

Résumé

L'objectif premier des systèmes de contrôle tolérant les défaillances (FTC) est de manier les défaillances ainsi que les divergences en utilisant des lois d'accomodation appropriés. L'objet d'obtenir des informations sur divers paramtres et signaux, qui doivent être manipulés dans le cadre de détection de défaillances, devient une tâche rigoureuse en fonction de l'augmentation du nombre de sous-systèmes. L'approche structurelle présentée dans ce raport, représente un cadre général pour fournir des informations quand le système devient complex. En outre, pour l'utilisation de cette approche, on peut determiner la séquence de calcul des résidues.

La méthodologie de cette approche est illustrée sur un bateau a propulsion "benchmark".

keywords Analyse structurelle, relations de redondance analytique, detection de défaillances et isolation.

Abstract

The prime objective of Fault-tolerant Control (FTC) systems is to handle faults and discrepancies using appropriate accommodation policies. The issue of obtaining information about various parameters and signals, which have to be monitored for fault detection purposes, becomes a rigorous task with the growing number of subsystems. The structural approach, presented in this paper, constitutes a general framework for providing information when the system becomes complex. Furthermore, by using this approach, one can determine the calculation sequences of the residuals.

The methodology of this approach is illustrated on the ship propulsion benchmark.

keywords   Structural analysis, analytical redundancy relations, fault detection and Isolation.

Contents

Chapter 1

Introduction

At the early stage of the design phase of a Fault-Tolerant Control (FTC) system for the process at hand, there is a need for early assessment of detectability (and if possible isolability) of the faults which are of interest. These faults can impose serious effects on the continuation of the process operation. This goal can not be achieved in the early stage by using analytical redundancy due to the lack of detailed information about the subsystems, for instance, lack of knowledge about the actual values of the involved parameters involved or parameter variations, and so on. However, it will be highly advantageous to be able to assess, at any stage of the design and with the available information, which part of the system can not be used for detecting the interesting fault(s) and which part of the system can possibly be used for that purpose.

The structural approach is a powerful method to achieve the above-mentioned goal. The *structural analysis* ([CLCS94], [SD27], [DS03]) is the analysis of the structural properties of the models, i.e., properties which are independent of the actual values of the parameters. Only links between the variables and parameters which are resulted from the operating model are represented in this analysis. They are independent from the operating model and are thus independent of the form under which this operating model is expressed (qualitative or quantitative data, analytical or non-analytical relations). The links are represented by a graph, on which a structural analysis is performed. In relation with this analysis only course initial data are needed.

As the basic knowledge about the process/model is available in form of variables, parameters, and some initial data, the structural analysis has following properties/advantages:

- It can be applied to linear as well as non-linear models.

- It does not need any kind of model accuracy.

- It allows a homogeneous representation of different models by means of the incidence matrix of a di-graph.

The report is formed in following chapters:

Chapter 1: Introduction.

Chapter 2: System description. The model of the ship propulsion benchmark, used in this report, constitutes of two parts, the dynamic part and the static one. The former is mainly a linear version of the benchmark while the later is represented by tables. As the result, the obtained overall model has maintained the complexity of the plant. It should be mentioned that the used model is obtained from the version of the benchmark which constitutes one engine + one propeller. Limitations (except one) are not considered at the first stage. The reason behind it is, that in the beginning of the design stage, there are no specific knowledge about the limitations/constraints in the system. So, having this lack of knowledge, it would be advantageous to know whether the actual configuration is informative enough for detection (and perhaps isolation) purposes.

Chapter 3: Structural Model and Canonical Decomposition The theory behind the structural approach is described using examples mainly from the ship benchmark. This approach, using well-known

di-graph theory, constitutes a general framework which shows its advantage when the number of components/subsystems in a complex system increases. Moreover, using the structural analysis, one can determine the calculation sequences of the residuals. The theory has been directly applied on the ship benchmark in the same chapter. The soul purpose for doing it is to illustrate the stepwise implementation of the theory on the system. Using canonical decomposition lead to obtaining residual expressions for the ship benchmark at the end of this chapter.

Chapter 4: Fault Detection and Isolation The resulting Analytical Redundancy Relations (ARRs) obtained by applying the approach are then considered in this chapter. Appropriate algorithms are used to detect the faults, and the results are then discussed.

Chapter 5: Conclusion An overall conclusion of the structural analysis method applied on the propulsion system is given in this chapter.

Chapter 2

Ship Benchmark

The chosen propulsion system, in this report, consists of one engine and one propeller. The original propulsion system consists of two engines + two propellers which are placed in parallel. However, the chosen system has the same performance as the original one. The idea is to use this system in order to understand the details and implemented functions and to try to design detector algorithms for it. Modification/extension/redesign of FDI algorithms to detect the same faults in the original system, will afterward be needed, since the engine/propeller sets in the original system are cross-coupled.

2.1 Description of the model

An outline of the propulsion system is drawn in figure 2.1. The dynamic part of the model is mainly a linear version of the benchmark. The static part of the model consists of three different tables of real data. The consequent overall model, therefore, maintains its complexity and nonlinearity of the model propulsion system.

The dynamic and static part of the benchmark are detailed in the following by referring to [IZB97], and fig. 2.1:

2.1.1 Propeller pitch angle control loop

The linearized version of the pitch angle control system is described by following equations:

$$u_c = k_t \left(\theta_{ref} - \theta_m \right) + (\Delta \dot{\theta}_{inc})$$
$$u_c = \dot{\theta}$$
$$\theta_m = \theta + \nu_\theta + (\Delta \theta)$$

k_t is the proportional controller, θ_{ref} is the reference signal, θ_m is the measured signal, and u_c is the control signal. $\Delta \dot{\theta}_{inc}$ and $\Delta \theta$ are incipient correspondingly sensor fault imposed on this system. ν_θ represents the measurement noise.

2.1.2 Governor

Input to the Governor, which is a PI controller, is the difference between the shaft speed reference n_{ref}, and the measured shaft speed n_m. The output is the fuel index Y_c. The PI controller is described by following functions:

$$
\begin{aligned}
n_m &= n + \nu_n (+\Delta n_m) \\
\dot{Y}_{PI} &= \frac{k_r}{\tau_i} [(n_{ref} - n_m) + \tau_i \frac{d\left(n_{ref} - n_m\right)}{dt}] \\
Y_{PI} &= \int \dot{Y}_{PI}.dt + Y_{PI_0} \\
Y_c &= \max(0, \min(Y_{PI}, 1)) \\
Y_m &= Y_c + \nu_Y
\end{aligned}
$$
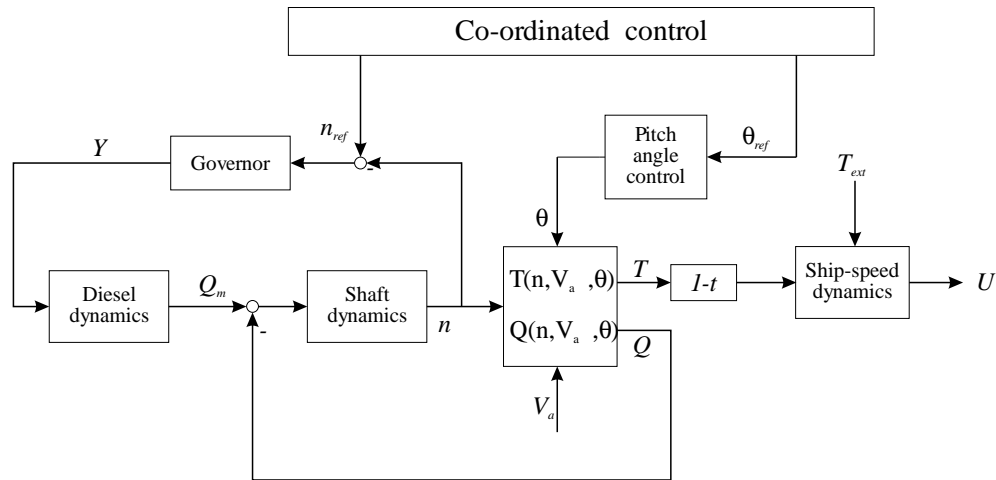
Figure 2.1: Diagram of the ship speed propulsion system

The dependence of the governor to the shaft speed is disregarded. Furthermore, in the real governor there exists anti wind-up arrangements, which are disregarded in this report. Y_{PI0} is the initial value and depends of the operating point of the system. Y_m is the measured fuel index. The term ν_Y represents the fact that the calculated fuel index by the controller is transformed to physical actions by some actuators (motors) and these generate some noises.

2.1.3 Diesel engine dynamics

The diesel engine dynamics can be divided in two parts. The first part, which describes the relation between generated torque and the fuel index, is in linearized form given by following ordinary (linear) differential equation:

$$k_y = k_{yc}(+\Delta k_y)$$
$$Q_m + \tau_c \frac{dQ_m}{dt} = k_y.Y_c \iff Q_m = ce^{-\frac{t}{\tau_c}} + k_y.Y_c$$

k_{yc} is the diesel engine's gain constant, τ_c is the time constant. The value of arbitrary parameter c is calculated by inserting initial values of the ODE.

The second part is derived, when considering the applied torques to the shaft, by following differential equation:

$$I_m \dot{n} = Q_m - Q - Q_f$$

Q_m [Nm] is the torque developed by the diesel engine, Q [Nm] is the developed torque from propeller dynamics, and Q_f [Nm] is the friction torque (not considered in this report).

The shaft speed can then be calculated by:

$$n = \int \dot{n}.dt + n_0$$

where n_0 in the initial value.

2.1.4 Propeller Characteristics

There are two types of propellers: fixed pitch propellers (FP), where the pitch angle is fixed, and controllable pitch propellers (CP), where the pitch angle θ can be changed. The pitch angle is the angle between a single propeller blade and the direction it turns. The pitch angle can be changed from 100% (full ahead) to -100% (full astern).

The CP-propeller characteristics are represented in the benchmark by two tables of real data: the first table characterizes the developed (consumed) propeller torque Q, and the second table characterizes the developed (produced) propeller thrust T:

$$f_Q(n, \theta, V_a, Q) = 0 : Table1$$
$$f_T(n, \theta, V_a, T) = 0 : Table2$$

where V_a (m/s) is the velocity of the water that goes into propeller disc (also called the advance speed). The magnitude of the advance speed V_a is less than the ship speed U. This is due to the non-uniform shape of the water flow under the ship hull. The water flow changes from the ship speed U under the ship to zero far behind the ship. The velocity of the water at the propeller, V_a, is therefor smaller than the ship speed. The influence on the advance speed can be described by the wake fraction numbers w, in the following equation:

$$V_a = (1 - w)U$$

Typical values for w are between 0.1 and 0.4. In addition to the wake another effect, the thrust deduction, has to be mentioned. Due to propeller movement, the water flow in the area behind the ship is increased. This disturbs the pressure balance between the front part of the ship, the bow, and the back part of the ship, the stern. This disturbance causes an additional resistance to the ship, i.e. a part of the thrust produced by the propeller is lost due to this additional resistance. This effect can be described by the thrust deduction number $(1 - t_T)$. t_T has typical values between 0.05 and 0.2.

$$T_{res} = (1 - t_T)T$$

2.1.5 Ship speed dynamics

Following equations describe the ship speed dynamics:

$$
\begin{aligned}
m.\dot{U} &= R_U + T_{res} \\
U &= \int \dot{U}.dt + U_0 \\
U_m &= U + \nu_U \\
R_U &= R(U) : Table3
\end{aligned}
$$

The hull resistant given by a table of real data, $R(U)$, describes the resistance of the ship in the water and is a negative quantity. ν_U is the measurement noise. m is the mass of the ship.

Chapter 3

Structural Model and Canonical Decomposition
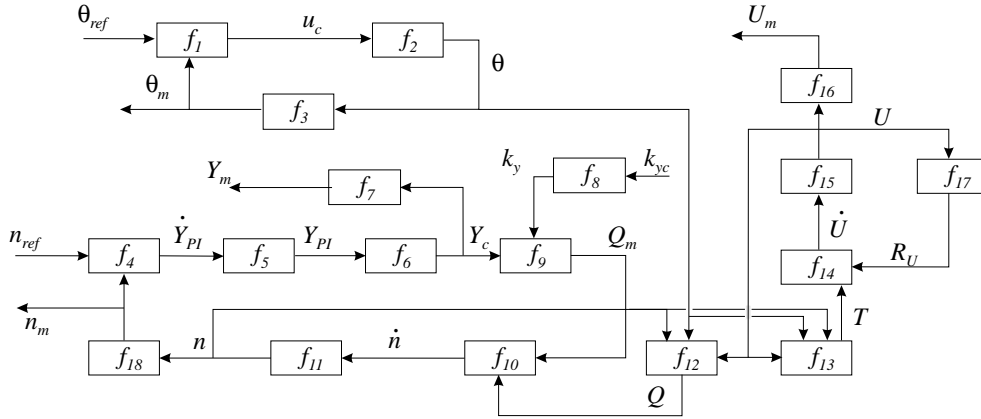
Figure 3.1: A graphical representation of the benchmark

Detailed model of large scale industrial plants are often not available. Control engineers are in practice provided by some graphical representations of the system which show the interconnections of a given number of blocks, each representing a subsystem or component (tank, pump, valve,...). These subsystems/components are described by different levels of available knowledge, for instance, transfer functions, equations, rules, tables, etc..

In addition to the data describing the plant, it is also equipped with a given set of sensors. These sensors are used for control, safety, and management purposes. They define, among the set of all variables which characterize the evolution of the plant, those which are known. Figure 3.1 shows a graphical representation of the benchmark (one engine + one propeller).

3.1 Defining the structure of the model

The model of the system is considered as a set of constraints which are applied to a set of known and unknown variables. The selected set of variables describes the evolution of the process.

The known variables are, sensor measurements, control variables (signals), variables with known values (constant, parameters), and reference variables (signals).

The set of constraints is given by the models of the blocks, see fig. 3.1, which constitute the system. The term "constraint" refers to the fact that a technological unit imposes some relations between the values of the variables, so that taking any possible value in the variable space except those compatible with the physical laws applied to that technological unit, is not possible. As an example [SD27], a resistor R, as a technological unit, introduces a constraint on the vector of variables (u, i) since their value can only lie in

the subspace defined by Ohm's law, i.e. $u - Ri = 0$.

The structure of the model is a di-graph which incidence matrix represents the links between variables (known and unknown) and the constraints.

Lets define following sets:

$\mathcal{F} = \{f_1, f_2, \cdots, f_m\}$ Set of constraints representing the system's model

$\mathcal{Z} = \overline{X} \cup X = \{z_1, z_2, \cdots, z_n\}$ Set of variables

$\overline{X}$ Set of known variables

X Set of non-measured variables (unknown variable)

The set of known variables can itself consists of different sets as outlined bellow

$\overline{X} = U \cup Y \cup C$ Set of known variables

U Set of control variables

Y Set of measured variables

C Set of variables with known values (f.x. constant parameters)

The structure of the model is described by the following binary relation:

$$S : \mathcal{F} \times \mathcal{Z} \longrightarrow \{0, 1\}$$

$$(f_i, z_j) \longrightarrow \begin{cases} S(f_i, z_j) = 1 & \text{iff the constraint } f_i \text{ applies to the variable } z_j, \\ S(f_i, z_j) = 0 & \text{Otherwise.} \end{cases}$$

Using the above mentioned definitions, for the simple model of the propulsion system, we get the following sets:

$$U = \{n_{ref}, \theta_{ref}, Y_m\}$$
$$Y = \{n_m, \theta_m, U_m\}$$
$$C = \{k_{yc}\}$$
$$X = \{n, \dot{n}, \theta, Y_{PI}, \dot{Y}_{PI}, Y_c, u_c, k_y, U, \dot{U}, Q_m, Q, T, R_U\}$$

and

$$\mathcal{Z} = \overline{X} \cup X = C \cup U \cup Y \cup X$$

and the constraints are defined as:

$f_1(\theta_{ref}, \theta_m, u_c) = 0$	:	$u_c = k_t * (\theta_{ref} - \theta_m)$	Pitch controller dynamics!
$f_2(u_c, \theta) = 0$	:	$u_c = \frac{d\theta}{dt} \quad \Leftrightarrow \quad \theta(t) = \int u_c dt + \theta_0$	Propeller pitch dynamics
$f_3(\theta_m, \theta) = 0$	:	$\theta_m = \theta$	Sensor measurement
$f_4(n_{ref}, n_m, \dot{Y}_{PI}) = 0$	:	$\dot{Y}_{PI} = \frac{k_r}{\tau_i}[(n_{ref} - n_m) + \tau_i \frac{d(n_{ref} - n_m)}{dt}]$	Shaft controller dynamics
$f_5(\dot{Y}_{PI}, Y_{PI}) = 0$	:	$Y_{PI} = \int \dot{Y}_{PI} dt + Y_{PI0}$	Shaft controller dynamics
$f_6(Y_{PI}, Y_c) = 0$	:	$Y_c = \max(0, \min(Y_{PI}, 1))$	Shaft controller dynamics
$f_7(Y_c, Y_m) = 0$	:	$Y_m = Y_c$	Sensor measurement
$f_8(k_y, k_{yc}) = 0$	:	$k_y = k_{yc}$	Diesel engine dynamics
$f_9(Y_c, k_y, Q_m) = 0$	:	$Q_m + \tau_c \frac{dQ_m}{dt} = k_y Y_c$	Diesel engine dynamics
$f_{10}(Q_m, Q, \frac{dn}{dt}) = 0$	:	$I_m \frac{dn}{dt} = Q_m - Q$	Shaft speed dynamics
$f_{11}(\frac{dn}{dt}, n) = 0$	:	$n = \int \dot{n} dt + n_0$	Shaft speed dynamics
$f_{12}(n, \theta, (1 - w)U, Q) = 0$	:	Table1	Developed propeller torque
$f_{13}(n, \theta, (1 - w)U, T) = 0$	:	Table2	Developed propeller thrust
$f_{14}(T, R_U, \frac{dU}{dt} = 0)$	:	$m \frac{dU}{dt} = T - R_U$	Ship speed dynamics
$f_{15}(U, \frac{dU}{dt} = 0)$	:	$U = \int \dot{U} dt + U_0$	Ship speed dynamics
$f_{16}(U, U_m) = 0$	:	$U_m = U$	Sensor measurement
$f_{17}(R_U, U) = 0$	:	Table3	Hull characteristics
$f_{18}(n, n_m) = 0$	:	$n_m = n$	Sensor measurement

The structural representation can be shown in a matrix format. This is illustrated in figure 3.2

	Y_m	n_m	U_m	θ_m	n_{ref}	θ_{ref}	k_{yc}	u_c	$\dot{Y}_{PI}$	Y_{PI}	Y_c	Q_m	$\dot{n}$	n	θ	k_y	$\dot{U}$	U	Q	T	R_U
f_1				1		1		1													
f_2								1								1					
f_3				1												1					
f_4		1			1				1												
f_5									1	1											
f_6										1	1										
f_7	1								1												
f_8							1									1					
f_9											1	1				1					
f_{10}												1	1						1		
f_{11}													1	1							
f_{12}													1	1			1	1			
f_{13}													1	1				1		1	
f_{14}																	1			1	1
f_{15}																	1	1			
f_{16}			1															1			
f_{17}																		1			1
f_{18}		1												1							

.

Figure 3.2: The structural representation of the benchmark in matrix format (incidence matrix)

3.1.1 Direct redundancy relations

Suppose that there exists a subset of $\mathcal{F}$, F such that:

$$\forall\, f \in F\ ,\ \ \forall\, x \in X:\quad S(f,x) = 0$$

Let $\mathcal{F}_k$ be the largest set which is characterized by this property.
This property can be understood as following: any constraint f in $\mathcal{F}_k$ is only applied to known variables and parameters and no unknown variable is involved. Each constraint, hence, constitutes a redundancy relation which can be checked on-line for FDI and monitoring purposes.
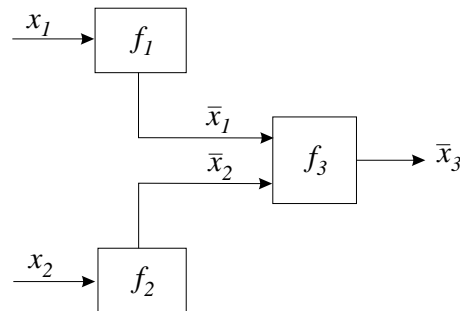
Figure 3.3: An example of direct redundancy relation

Example1: Since in figure 3.3 $\overline{X} = \{\overline{x}_1, \overline{x}_2, \overline{x}_3\}$, the constraint f_3 becomes a direct redundancy relations as it follows the abovementioned definition. f_1 and f_2 can represent sensors while f_3 can represent a

linear equation with known parameters.

The set $\mathcal{F}_k$, therefore, constitutes a set of direct redundancy relations which are obtained in a straightforward way from the model of the plant and the instrumentation.

The complement set of $\mathcal{F}_k$ in $\mathcal{F}$ is denoted $\mathcal{F}_x$, i.e.

$$\mathcal{F}_k \cup \mathcal{F}_x = \mathcal{F}$$

3.1.2 Deduced redundancy relations

Let $\mathbb{P}(\mathcal{E})$ denote the set of the subsets of a set $\mathcal{E}$. The rows and columns structures are then defined by following equations:

$$
\begin{aligned}
\mathbb{P}(\mathcal{F}_x) & \xrightarrow{Q} \mathbb{P}(\mathcal{Z}) \\
F & \longrightarrow Q(F) = \{z_j \mid \exists f_i \in F, \text{ such that } S(f_i, z_j) = 1\} \\
\mathbb{P}(\mathcal{Z}) & \xrightarrow{R} \mathbb{P}(\mathcal{F}_x) \\
Z & \longrightarrow R(Z) = \{f_j \mid \exists z_i \in Z, \text{ such that } S(f_j, z_i) = 1\}
\end{aligned}
$$

Example 2: Lets choose a set of constraints (row structure) $F(\subset \mathcal{F}_x) = \{f_1, f_2, f_3, f_9\}$, then by definition $Q(F) = \{u_c, \theta, \theta_m, \theta_{ref}, Q_m, k_y, Y_c\}$. Similarly, if we choose a set of variables (column structure) $Z(\subset \mathcal{Z}) = \{n_m, U, \theta\}$, by definition we get $R(Z) = \{f_2, f_3, f_4, f_{12}, f_{13}, f_{15}, f_{16}, f_{17}, f_{18}\}$.

3.1.3 Subsystems characterization

Definition: A subsystem is a pair $(F, Q(F))$, where F is a subset of $\mathcal{F}_x$. Let

$$Q(F) = Q_k(F) \cup Q_x(F)$$

where Q_k and Q_x are the subset of the known, correspondingly unknown variables. Then the constraints which define the subsystem are written as:

$$F(Q_k(F), Q_x(F)) = 0$$

The subsystem F is characterized based on the number of existing solutions for $Q_x(F)$, given the known values (or trajectories) in $Q_k(F)$.

Definition: The subsystem F is said to be *compatible* if, for any given value of $Q_k(F)$, the set of the values of $Q_x(F)$ which satisfy the constraints F is not empty.

F is *under-determined* if, for any given value of $Q_k(F)$, the set of values of $Q_x(F)$ which satisfy the constraint F is of cardinal larger than 1 and it is *determined* if the cardinal is equal to 1.

A determined system is said to be *over-determined* if

$$
\begin{aligned}
\exists \Phi \subset \quad & F \quad \text{and} \quad \Phi \neq F \quad \text{such that} \\
Q_x(\Phi) = \quad & Q_x(F)
\end{aligned}
$$

and for any of the variables $Q_k(F)$ the values of $Q_x(\Phi)$ which satisfy the constraints Φ are the same as those of $Q_x(F)$ which satisfy the constraints F.

A determined system which is not over-determined is said to be *just-determined*.

The interpretation of over, just, and under-determined subsystems is given in the following.

- If the subsystem $(F, Q_k(F), Q_x(F))$ is under-determined, then there exists several solutions for the variables in $Q_x(F)$. Therefore, they can not be computed using the known values in $Q_k(F)$ and the constraints in F. Under-determined subsystems exist due to existence of unobservable variables or insufficient knowledge about the system/process which is reflected into the existing model.
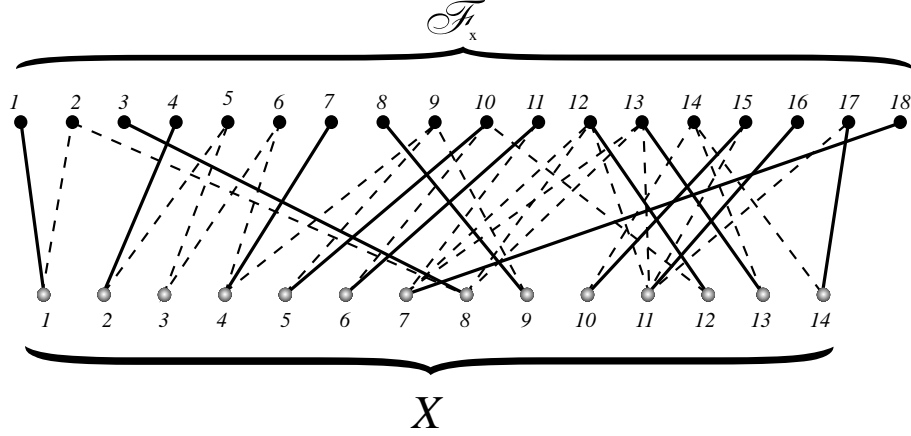
Figure 3.4: A Possible matching on the graph representing the ship control system's structure

- The subsystem $(F, Q_k(F), Q_x(F))$ is just-determined $\Leftrightarrow$ there exists a unique solution for the variables in $Q_x(F)$, given known values $Q_k(F)$ and constraints F.

- The subsystem $(F, Q_k(F), Q_x(F))$ is over-determined $\Leftrightarrow$ there exists a unique solution for the variables in $Q_x(F)$ **and** this solution can be computed in several ways by using the known values $Q_k(F)$ and constraints F. Referring to the definition above, each subset $\Phi \subset F$ satisfying the definition provides a mean for computation of $Q_x(F)$. Since all the results have to be the same, one obtains deduced redundancy relations by writing the coherence conditions.

The next section shows how the structural analysis of graphs is used to find these three classes of subsystems.

3.2 Canonical Decomposition

As mentioned before, the problem of finding the redundancy relations which can be obtained in a given system is equivalent to the problem of finding the over-determined subsystems in the structure of the system.

3.2.1 System's structural graph

The structure of the system is represented by a di-graph. Recalling that $\mathcal{F}$, represents the system (manifested by its relations) and $\mathcal{Z}$ are the variables in the system, a di-graph associates the sets $\mathcal{F}$ and $\mathcal{Z}$ with a set of links between their elements $A_\mathcal{Z}$, i.e.

$$(f, z) \in A_\mathcal{Z} \iff S(f, z) = 1$$

3.2.1.1 Complete matching

Consider a graph $\mathfrak{G}(F_X, X, A_X)$ representing a restricted part of the system's structural graph to the set of vertices X (for the variables) and $F_X = R(X)$ (for the constraints). A_X then represents the arcs connecting (linking) F_X to X.
Let a belong to A_X, then $X(a)$ is called the *extremity* of a in X and $F_X(a)$ is called the extremity of a in F_X, and a can be written as:

$$a = (F_X(a), X(a))$$

in another words, the arc (or link) a is defined by a pair, where the first element is a constrict and the second element is an unknown variable and these two are connected through a.

Definition: $\mathfrak{G}(F_X, X, A)$ is a _matching_ on $\mathfrak{G}(F_X, X, A_X)$ iff:

1) $A \subset A_X$,

2) $\forall \; a1, a2 \in A \; | \; a1 \neq a2 \Leftrightarrow F_X(a1) \neq F_X(a2) \; \wedge \; X(a1) \neq X(a2)$

A possible matching on the graph representing ship control system's structure is shown in figure 3.4.

Definition: $\mathfrak{G}(F_X, X, A)$ is a _maximal matching_ on $\mathfrak{G}(F_X, X, A_X)$ iff:

$$\forall \; A' \supset A \; | \; A' \neq A \; : \; \mathfrak{G}(F_X, X, A') \quad \text{is not a matching}$$

Definition: $\mathfrak{G}(F_X, X, A_X)$ is _complete_ with respect to F_X (respectively with respect to X) iff:

$$\forall \; f \in F_X \; \exists \, a \in A \; : \; F_X(a) = f,$$
$$(\text{resp.} \quad \forall \; x \in X \quad \exists \, a \in A \; : \; X(a) = x \quad)$$

The basic condition for the existence of complete matching is given by the Könenig-Hall theorem:

Theorem [refer]: _A complete matching with respect to_ F_X _exists on_ $\mathfrak{G}(F_X, X, A_X)$ _iff:_

$$\forall F' \subset F_X \quad |Q_X(F')| \geq |F'|$$

A dual result is also stated for the existence of a complete matching with respect to X.

3.2.1.2 Decomposition

The pair $(\alpha, \; \beta)$ is an _external support_ of $\mathfrak{G}(F_X, X, A_X)$ iff:

1) $\alpha \subset F_X \; \wedge \; \beta \subset X$

2) $A_X \cap (\bar{\alpha}, \; \bar{\beta}) = \emptyset$

where $\bar{\alpha}$ (resp. $\bar{\beta}$) is the complement of α in F_X (resp. β in X).
Let Γ_X be the set of external supports of $\mathfrak{G}(F_X, X, A_X)$. The external _dimension_ of $\mathfrak{G}(F_X, X, A_X)$ is defined by:

$$\dim(\mathfrak{G}(F_X, X, A_X)) = \min_{((\alpha, \; \beta) \in \Gamma_X)} (|\alpha| + |\beta|)$$

Minimal external supports are those external supports for which the minimum is obtained (or the dimension is obtained).

Theorem [refer]: _For any di-graph of finite external dimension, there exist two uniquely defined external supports_ $(\alpha_*, \; \beta^*)$ _and_ $(\alpha^*, \; \beta_*)$ _such that, if_ $(\alpha, \; \beta)$ _is another minimal external support, then:_

1) $\alpha_* \subset \alpha \qquad$ or $\quad \alpha_* = \emptyset$

2) $\alpha \subset \alpha^*$

3) $\beta_* \subset \beta \qquad$ or $\quad \beta_* = \emptyset$

4) $\beta \subset \beta^*$

Definition: The three canonical components of the di-graph $\mathfrak{G}(F_X, X, A_X)$ are defined by:

$$\begin{aligned}
B^+ &= \mathfrak{G}(\overline{\alpha^*}, \; \beta_*, \; A_X^+) \\
B^- &= \mathfrak{G}(\alpha_*, \; \overline{\beta^*}, \; A_X^-) \\
B^0 &= \mathfrak{G}(\alpha^* \backslash \alpha_*, \; \beta^* \backslash \beta_*, \; A_X^0)
\end{aligned}$$

where

$$
\begin{aligned}
A_X^+ &= A_X \cap (\overline{\alpha^*},\, \beta_*) \\
A_X^- &= A_X \cap (\alpha_*,\, \overline{\beta^*}) \\
A_X^0 &= A_X \cap (\alpha^* \backslash \alpha_*,\, \beta^* \backslash \beta_*)
\end{aligned}
$$

A canonical decomposition of the whole structure of the system including the di-graph $\mathfrak{G}(F_X, X, A_X)$ and the known variables is shown in figure (3.5). An interpretation of the three canonical components are given in following subsection.
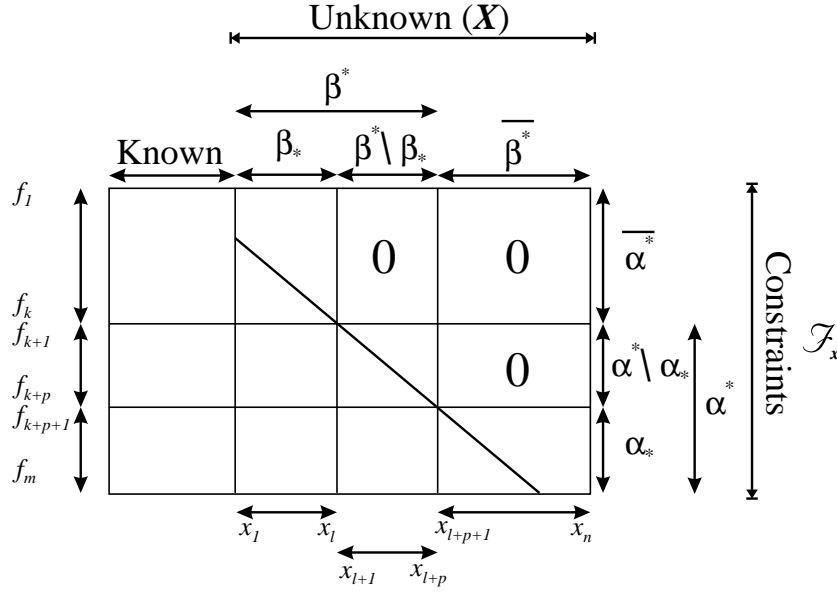

Figure 3.5: Canonical decomposition of the system into under-, just-, and over-determined subsystems.

3.2.1.3   Interpretation

A matching is a set of pairs (f_j, x_i) such that any constraints $f_j \in F_X$ (resp. any variable $x_i \in X$) belongs at most to one pair. This leads to following hypothesis:

Hypothesis: *The unknown variable x_i is computed using the constraint f_j, under the assumption that all other variables $Q(fj) \backslash xi$ are known.*

Consider a complete matching with respect to X' and F', where $X' \subset X$ and $F' \subset \mathcal{F}_x$. As the matching with respect to both X' and F' is complete then by definition following equation is valid (true):

$$
|F'| = |X'|
$$

This statement/equation can be interpreted as following: The unknown variables in the set X' can be computed by using the constraints in set F' under the assumption that all other variables are known. In the case of numerical analysis, this represents the case of solving a system of n equations with n unknown variables which has "almost" always a solution [1].

3.2.1.4   Subsystem interpretation

Referring to figure 3.5 the subsystem $(\overline{\alpha^*},\, \beta_*)$ appears to be over-determined, as the maximum matching is complete with respect to the variables but not with respect to the constraints. Expressing it in another way, the number of constraints exceeds the number of unknown variables, i.e.:

$$
\overline{\alpha^*} = \{f_1, \cdots, f_k\}\ ,\ \beta_* = \{x_1, \cdots, x_l\}\ \wedge\ k > l
$$

---

[1] The existence of the solusion depends on the values of the parameters in the involved relations.

The variable β_* can be computed using a subset of the constraints $\overline{\alpha^*}$ and the values of the known variables and then substituted into the remaining constraints in order to obtain deduced redundancy relations.

Since different complete matchings with respect to the variables can be constructed, there are different possible sets of deduced redundancy relations.

The subsystem $(\overline{\alpha^*}, \beta_*)$ constitutes the monitorable part of the system (for a given partition into known and unknown variables) since it presents two fundamental properties:

1. It is *computable*: the variables β_* can be deduced from the known variables.

2. It is *redundant*: some coherence relations (i.e. the deduced redundancy relations) must hold.

The subsystem $(\alpha^* \backslash \alpha_*, \beta^* \backslash \beta_*)$ is a just-determined subsystem, as the maximum matching is complete with respect to both the variables and the constraints, i.e.:

$$\alpha^* \backslash \alpha_* = \{f_{k+1}, \cdots, f_{k+p}\} \ , \ \beta^* \backslash \beta_* = \{x_{l+1}, \cdots, x_{l+p}\}$$

Having knowledge about the known variables and the variables in β_*, the variables in the subset $\beta^* \backslash \beta_*$ can be calculated by solving the constraints $\alpha^* \backslash \alpha_*$. Notice that the obtained solution will be unique.

Since no redundancy does exists, the subsystem can not be monitorable, i.e. a malfunction which would occur within any of the constraints $\alpha^* \backslash \alpha_*$, or in any sensor providing a known variable y such that $R(y) \subset \alpha^* \backslash \alpha_*$ could not be detected. $R(y)$ is the set of relations which contain the known variable y.

The subsystem $(\alpha_*, \overline{\beta^*})$ is under-determined, since the maximum matching is complete with respect to the constraints but not the variables, i.e.:

$$\alpha_* = \{f_{k+p+1}, \cdots, f_m\} \ , \ \overline{\beta^*} = \{x_{l+p+1}, \cdots, x_n\} \ \wedge \ m - k < n - l$$

This means that the system is neither computable nor monitorable as the number of constraints is clearly less than the number of variables.

3.3 Matching under Causal Constraints

Using the definition for matching one can state that if $\mathfrak{G}(F_X, X, A)$ is a matching on $\mathfrak{G}(F_X, X, A_X)$ then $A \subset A_X$, which means that if $(f_j, x_i) \in A$ then $S(f_j, x_i) = 1$ or in another words x_i belongs to the structure of f_j. However, it does not by itself guarantee that the hypothesis in subsection 3.2.1.3 is valid: $x_i \in Q(f_j)$ is a necessary but not a sufficient condition. This is illustrated by following simple example from the benchmark:

Example: Consider the constraint $f_2(u_c, \theta) = 0 : u_c = \frac{d\theta}{dt}$ which shows the relation between the control signal u_c and pitch angle in propeller pitch control loop. Obviously, the control signal u_c can be computed by knowing the values of θ through constraint f_2. But the opposite is not true: θ can not be computed exactly using the values of u_c since θ is given by $\theta = \int u_c dt + \theta_0$ and the initial value θ_0 is not known (recall that $\theta \in X$). The possible and impossible matching is shown in figure 3.6. It also shows the matrix representation for the example. The $\times$ indicates the considered causality in the constraint.

This simple example shows that in order to use the interpretation in subsection 3.2.1.3 some calculability conditions has to be fulfilled (by the matching). This has led to introduction of causality in the structure of the system.

The causal graph for the systems' structure is a graph $\mathfrak{G}(F_X, X, A_X^c)$ such that:

1) $A_X^c \subset A_X$,

2) $(f_j, x_i) \in A_X^c$ *iff* x_i is computable in a unique way, using the constraint f_j,

under the assumption that all the other variables $Q(f_j) \backslash x_i$ are known.

a) Possible matching   b) Impossible matching

	u_c	θ
f_2	1	x

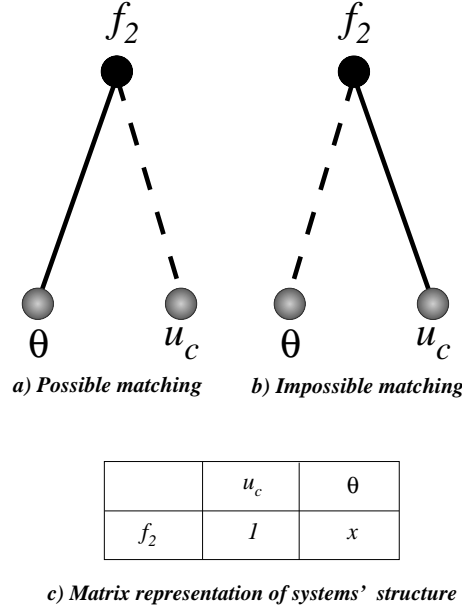c) Matrix representation of systems' structure .

Figure 3.6: An example of possible and impossible matching due to causality effect.

The causal matching on the graph $\mathfrak{G}(F_X, X, A_X)$ is defined as a graph $\mathfrak{G}(F_X, X, A)$ such that:

1) $A \subset A_X^c$,

2) $\forall \ a1, a2 \in A \mid a1 \neq a2 \Leftrightarrow F_X(a1) \neq F_X(a2) \ \wedge \ X(a1) \neq X(a2)$

A causal matching and causal complete matching will be defined as before, with the restriction $A \subset A_X^c$ in stead of $A \subset A_X$ and all previous results will hold.

Taking the causality into considerations for the ship benchmark, the structural representation can be shown in a matrix format shown in figure 3.7. The causality constraints are highlighted by replacing 1's by ×'s. The circles around 1's illustrate how the matching is been performed to obtain the di-graph representation in figure 3.4. These circles represent the solid lines.

It should be mentioned that the possible matching in figure 3.4 is in fact a causal matching as well.

3.3.1   Residual expressions for the ship propulsion benchmark

Taking the causality constraint f_6 under considerations, the structural model of the ship which is shown in figure 3.4, can be decomposed into an under-determined subsystem and an over-determined subsystem . The existence of the under-determined subsystem is due to constraints f_5 and f_6 where the unknown variable Y_{PI} can not be explicitly determined as function of variables Y_c and $\dot{Y}_{PI}$.

In the over-determined subsystem, three constraints: f_2, f_9, and f_{14} are not used to perform the matching and thus can be used to generate three ARRs.

Further simplifications   Additional examination of the system's dynamics shows that the normal operation of the system mainly lies on the linear part of the constraint f_6, i.e. $Y_c = Y_{PI}$. This leads to defining an additional ARR by replacing the dashed line (arc) connecting the third unknown variable in figure 3.4 to the sixth constraint with a solid line (arc). Then the constraint f_5 can be used to generate an additional ARR.

The way to obtain ARRs can be represented graphically as shown by the figure 3.8: the over-determined subsystem is further decomposed into smaller subsystems each representing one ARR. These graphs are oriented ones since they take into account calculability constraints which express that functions can or can not be inverted. The big dark circles in the figure represent the available data at any time, i.e. they are

	Y_m	n_m	U_m	θ_m	n_{ref}	θ_{ref}	k_{yc}	u_c	$\dot{Y}_{PI}$	Y_{PI}	Y_c	Q_m	$\dot{n}$	n	θ	k_y	$\dot{U}$	U	Q	T	R_U
f_1				*1*		*1*		①													
f_2								*1*							*x*						
f_3				*1*											①						
f_4		*1*			*1*				①												
f_5									*1*	*x*											
f_6										*x*	*1*										
f_7	*1*											①									
f_8							*1*									①					
f_9											*1*	*x*				*1*					
f_{10}													①	*1*					*1*		
f_{11}														①	*x*						
f_{12}															*x*	*x*		*x*	①		
f_{13}															*x*	*x*		*x*		①	
f_{14}																	*1*			*1*	*1*
f_{15}																	①	*x*			
f_{16}			*1*														①				
f_{17}																		*x*			①
f_{18}		*1*											①								

Figure 3.7: Matrix representation of the system's structure with causality constraints.

information sources, i.e. they belong to the set $\bar{X}$.

As mentioned before, the ARRs can be directly deduced from the decomposed (over-determined) sub-graphs sequentially. This is exemplified by considering one of the subgraphs, for instance case a) in figure 3.8. The two unknown variables u_c and θ can respectively be calculated using f_1 and f_3. The ARR expression is obtained using the function f_2:

$$\frac{d\theta_m}{dt} - k_t.(\theta_{ref} - \theta_m) = 0$$

A *residual* is the result of the calculation of an ARR when the known variables are replaced by their values. Two residual expressions can thus be deduced: the *calculation* form, which allows to calculate the residual value and the *evaluation* form, which allows to explain this obtained residual value. The calculation expressions for residuals $r_1, \cdots, r_4$, deduced from subgraphs $a), \cdots, d)$ in figure. 3.8 are:

$$r_1 \quad = \quad \frac{d\theta_m}{dt} - k_t.(\theta_{ref} - \theta_m)$$

$$r_2 \quad = \quad \frac{1}{m}(R_{U_m} - f_T(n_m, \theta_m, (1-w).U_m).(1-t_T)) + \frac{dU_m}{dt}$$

$$r_3 \quad = \quad \frac{dY_m}{dt} - \frac{k_r}{\tau_i}(n_{ref} - n_m + \tau_i \frac{d(n_{ref} - n_m)}{dt})$$

$$r_4 \quad = \quad \frac{d^2 n_m}{dt^2} + \frac{df_Q(n_m, \theta_m, (1-w).U_m)}{I_m dt} + \frac{f_Q(n_m, \theta_m, (1-w).U_m)}{I_m \tau_c} - \frac{k_y Y_m}{I_m \tau_c} + \frac{dn_m}{\tau_c dt}$$

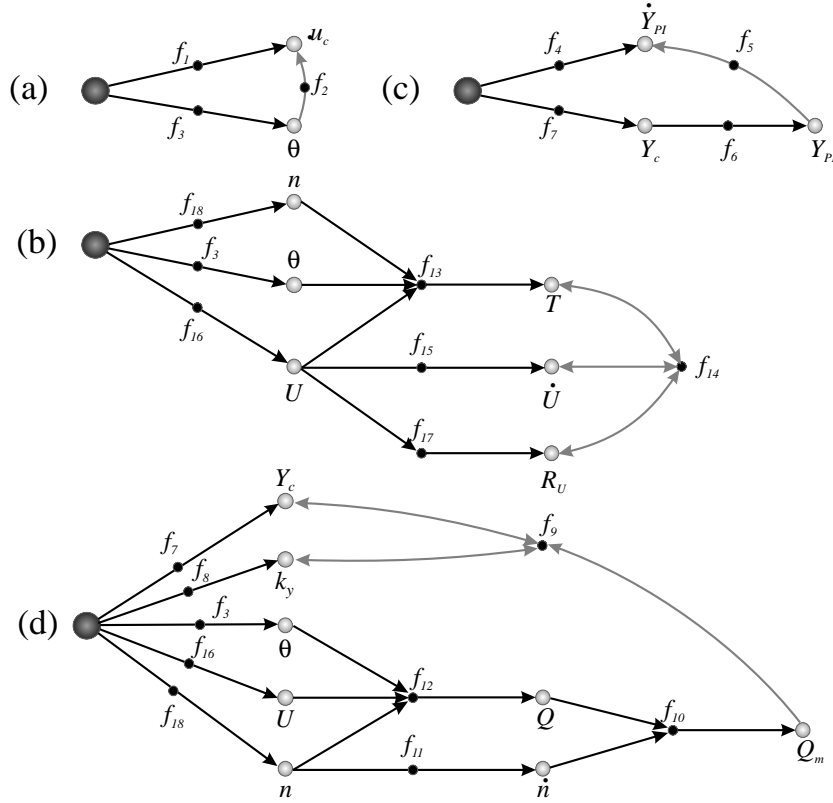where f_T, f_Q, and R_U represent three different tables.

Figure 3.8: The over-determined part of the system (represented by graphs)

A summary of evaluation form for the abovementioned residuals are given bellow:

$$r_1 = \frac{d\Delta\theta_m}{dt} + k_t\Delta\theta_m - \Delta\dot{\theta}_{inc}$$

$$r_2 = F_T(\Delta n_m, \Delta\theta_m, w, t_T)$$

$$r_3 = \frac{k_t}{\tau_i}(\Delta n_m - \tau_i\frac{d\Delta n_m}{dt})$$

$$r_4 = F_Q(\Delta n_m, \Delta\theta_m, w, \Delta k_y)$$

Using the evaluation form one can obtained the sensitivity expression of one residual to each fault by taking partial derivatives w.r.t. variables or parameters. However, because of the tables (functions f_{12}, f_{13}, and f_{17}, indicated by unknown numerical functions F_T and F_Q, these sensitivities can't be calculated for the residuals b) and d). Only simulation tests can be performed to evaluate these. An evaluation procedure has been carried out for the first residual:

$$\frac{dr_1}{d\theta_m} = \frac{d(\frac{d\theta_m}{dt})}{d\theta_m} + k_t$$

$$\frac{dr_1}{du_c} = -1$$

$$0 \qquad \text{otherwise}$$

The evaluation shows that the first residual is directly affected by incipient fault ($\Delta\dot{\theta}_{inc}$) and the sensor fault ($\Delta\theta_m$).

The table bellow shows which fault affects different residuals. The table also shows that some of the residuals are sensitive to parameter uncertainties. This existence of tables of real data prevents an analytical calculation to be carried out. It should be noticed that the sensitivity degree of residuals with respect to corresponding faults (and uncertain parameters) are not considered in this table.

	$\Delta\dot{\theta}_{inc}$	$\Delta\theta_m$	Δn_m	Δk_y	t_T	w
r_1	1	1				
r_2			1	1	1	1
r_3			1			
r_4		1	1	1		1

Table 3.1: Evaluation table showing the possible effect of faults and uncertainties on different residuals

Chapter 4

Fault Detection and Isolation

In this chapter two types of algorithms have been used for detection purposes. These are originally adapted from [BN94]. The one is originally an off-line *change detection in jump* method which is adapted for on-line fault detection. The other one is the known *generalized likelihood ratio test.* Two sets of signals have been tested: residuals where faults have small magnitude and residuals with faults with big magnitude. A conclusion is given at the end of this chapter. The residuals are iterated in the following:

$$r_1 = \frac{d\theta_m}{dt} - k_t.(\theta_{ref} - \theta_m)$$

$$r_2 = \frac{1}{m}(R_{U_m} - f_T(n_m, \theta_m, (1-w).U_m).(1-t_T)) + \frac{dU_m}{dt}$$

$$r_3 = \frac{dY_m}{dt} - \frac{k_r}{\tau_i}(n_{ref} - n_m + \tau_i \frac{d(n_{ref} - n_m)}{dt})$$

$$r_4 = \frac{d^2 n_m}{dt^2} + \frac{df_Q(n_m, \theta_m, (1-w).U_m)}{I_m dt} + \frac{f_Q(n_m, \theta_m, (1-w).U_m)}{I_m \tau_c} - \frac{k_y Y_m}{I_m \tau_c} + \frac{dn_m}{\tau_c dt}$$

where f_T, f_Q, and R_{U_m} represent three different tables.

4.1 Change detection

The problem of change detection is transformed into the following statistical problem. Lets consider a stochastic process (Y_t), with conditional distribution $p_\theta(y_t|y_{t-1}, \cdots, y_0)$. Given a record $(y_t)_{(0 \le t \le n)}$ decide between the two hypothesis:

$$\mathbb{H}_0 : \quad \theta = \theta_0 \tag{4.1}$$

and

$$\mathbb{H}_1 : \quad \exists r , \quad 1 \le r \le n \quad : \quad \begin{cases} \theta = \theta_0' & \text{for} \quad 0 \le r \le r-1 \\ \theta = \theta_1 & \text{for} \quad r \le r \le n \end{cases} \tag{4.2}$$

where the case $\theta_0' = \theta_0$ is often considered for on-line approach [BN94].

Let $(\epsilon_n)_n$ be a white noise sequence with variance σ^2, and let $(y_n)_n$ be the sequence of the residuals such that:

$$y_n = \mu_n + \epsilon_n \tag{4.3}$$

where:

$$\mu_n = \begin{cases} \mu_0 & \text{if} \quad n \le r-1 \\ \mu_1 & \text{if} \quad n \ge r \end{cases} \tag{4.4}$$

The problem is to detect the change in the mean μ_n, to estimate the change time r and possibly the mean values μ_0 and μ_1 before and after the jump. Of interest is the case where μ_1 is not known in advance and is most actual in practice:

4.1.1 Known means before and after jump

The detection problem consists of testing between the no change hypothesis:

$$\mathbb{H}_0 \quad : \quad r > n$$

and the change hypothesis:

$$\mathbb{H}_1 \quad : \quad r \leq n$$

The likelihood ratio between these two hypothesis is:

$$\prod_{k=r}^{n} \frac{p_1(y_k)}{p_0(y_k)} \tag{4.5}$$

where p_i is the Gaussian probability density with mean $\mu_i (i = 0, 1)$. The algorithm is thus:

$$\begin{aligned} \Lambda_n(r) &= \frac{\mu_1 - \mu_0}{\sigma^2} \sum_{k=r}^{n} \left(y_k - \frac{\mu_1 + \mu_0}{2} \right) \\ &= \frac{1}{\sigma^2} S_r^n(\mu_0, \nu) \end{aligned}$$

where

$$S_i^j(\mu, \nu) = \nu \sum_{k=i}^{j} \left(y_k - \mu - \frac{\nu}{2} \right) \tag{4.6}$$

and

$$\nu = \mu_1 - \mu_0$$

is the magnitude of the jump.

Replacing the unknown jump time r by its maximum likelihood estimate (MLE) under $\mathbb{H}_1$, namely:

$$\begin{aligned} \hat{r}_n &= \underset{1 \leq r \leq n}{\arg\max} \left[\prod_{k=0}^{r-1} p_0(y_k) \prod_{k=r}^{n} p_1(y_k) \right] \\ &= \underset{1 \leq r \leq n}{\arg\max} \ S_r^n(\mu_0, \nu) \end{aligned} \tag{4.7}$$

The following change detector is then obtained:

$$g_n \triangleq \Lambda_n(\hat{r}_n) = \max_r S_r^n(\mu_0, \nu) \underset{\mathbb{H}_0}{\overset{\mathbb{H}_1}{\gtrless}} \lambda \tag{4.8}$$

where λ is the threshold. In other words, decide $\mathbb{H}_1$ whenever g_n exceeds λ, and $\mathbb{H}_0$ otherwise.

Using Page-Hinkley stopping rule (also called cumulative sum algorithm), a jump in the mean at the first time n is detected when:

$$g_n = S_1^n(\mu_0, \nu) - \min_{1 \leq k \leq n} S_1^k(\mu_0, \nu) > \lambda \tag{4.9}$$

This equation can be recursively computed in following manner:

$$g_n = \left(g_{n-1} + y_n - \mu_0 - \frac{\nu}{2} \right) \tag{4.10}$$

It may be used more generally to detect any change between two known probability lows p_{θ_0} and p_{θ_1}. In this case we shall compute:

$$S_i^j(p_{\theta_0}, p_{\theta_1}) = \sum_{k=i}^{j} \log \frac{p_{\theta_0}(y_k)}{p_{\theta_1}(y_k)} \tag{4.11}$$

Further discussions on the theoretical properties and related subjects are discussed in [BN94] and the references herein.

4.1.2 Unknown jump magnitude

The more realistic case where the jump magnitude ν is unknown is considered in this section. Assume that μ_0 is known, but not μ_1. Two approaches may be used in such a case.

Approach 1: Two tests are run in parallel using equation (4.9), corresponding to an *a priori* chosen minimum jump magnitude ν_m and to two possible directions (increase or decrease in the mean). The corresponding stopping rules are as follows,
For an increase:

$$\begin{cases} T_0 = 0 \\ \\ T_n = \sum_{k=1}^{n} \left(y_k - \mu_0 + \frac{\nu_m}{2} \right) \\ \\ M_n = \max_{0 \le k \le n} T_k \\ \\ \text{alarm} \quad \text{when} \quad M_n - T_n > \lambda \end{cases} \tag{4.12}$$

and for an decrease:

$$\begin{cases} U_0 = 0 \\ \\ U_n = \sum_{k=1}^{n} \left(y_k - \mu_0 - \frac{\nu_m}{2} \right) \\ \\ m_n = \min_{0 \le k \le n} U_k \\ \\ \text{alarm} \quad \text{when} \quad U_n - m_n > \lambda \end{cases} \tag{4.13}$$

The decision is taken when the first alarm is generated by one of the stop rules (Equ. (4.12) or (4.13)). The estimate of the jump time r is the last maximum (respectively minimum) time before detection.

Approach 2: The unknown jump magnitude ν is replaced with its MLE. The likelihood ratio test is hence:

$$\max_{0 \le r \le n} \max_{\nu} S_r^n(\mu_0, \nu) \underset{\mathbb{H}_0}{\overset{\mathbb{H}_1}{\gtrless}} \lambda \tag{4.14}$$

Using Equation (4.6), we obtain:

$$\hat{\nu}_n(r) \triangleq \arg \max_{\nu} S_r^n(\mu_0, \nu) = \frac{1}{n-r+1} \sum_{k=r}^{n} (y_k - \mu_0) \tag{4.15}$$

On-line calculation of ν : The equation 4.15 can not be used in on-line approach as it is dependent of the jump time r which is not known in advance. To avoid this problem, the minimum jump magnitude v_m can be on-line detected based on a number, w, of past observations of the data (measurements). In other words, ν_m is detected on the basis of a moving window of the size w. This is computed in the following way:

$$
\begin{aligned}
\hat{\nu}_m(t)\Big|_w &= \frac{1}{w}\left(\sum_{k=t-w+1}^{t}\left(y(k)-\mu_0\right)\right) \\
&= \hat{\nu}_m(t-1)\Big|_w + \frac{1}{w}\left(y(t)-y(t-w)\right)
\end{aligned}
\tag{4.16}
$$

where t denotes the actual time instant. For the time instants less than the window's size, i.e. $t \in [1\cdots w]$, following method can be used:

$$
\begin{aligned}
\hat{\nu}_m(t) &= \frac{1}{t}\left(\sum_{k=1}^{t}\left(y(k)-\mu_0\right)\right) \\
&= \hat{\nu}_m(t-1) + \frac{1}{t}\left(y(t)-\mu_0-\hat{\nu}_m(t-1)\right)
\end{aligned}
\tag{4.17}
$$

with the initial value chosen as $\hat{\nu}(0) = \mu_0$.

4.1.3 Generalized likelihood ratio (GLR) test

(a) First residual (small signals)

(b) Second residual (small signals)

(c) Third residual (small signals)

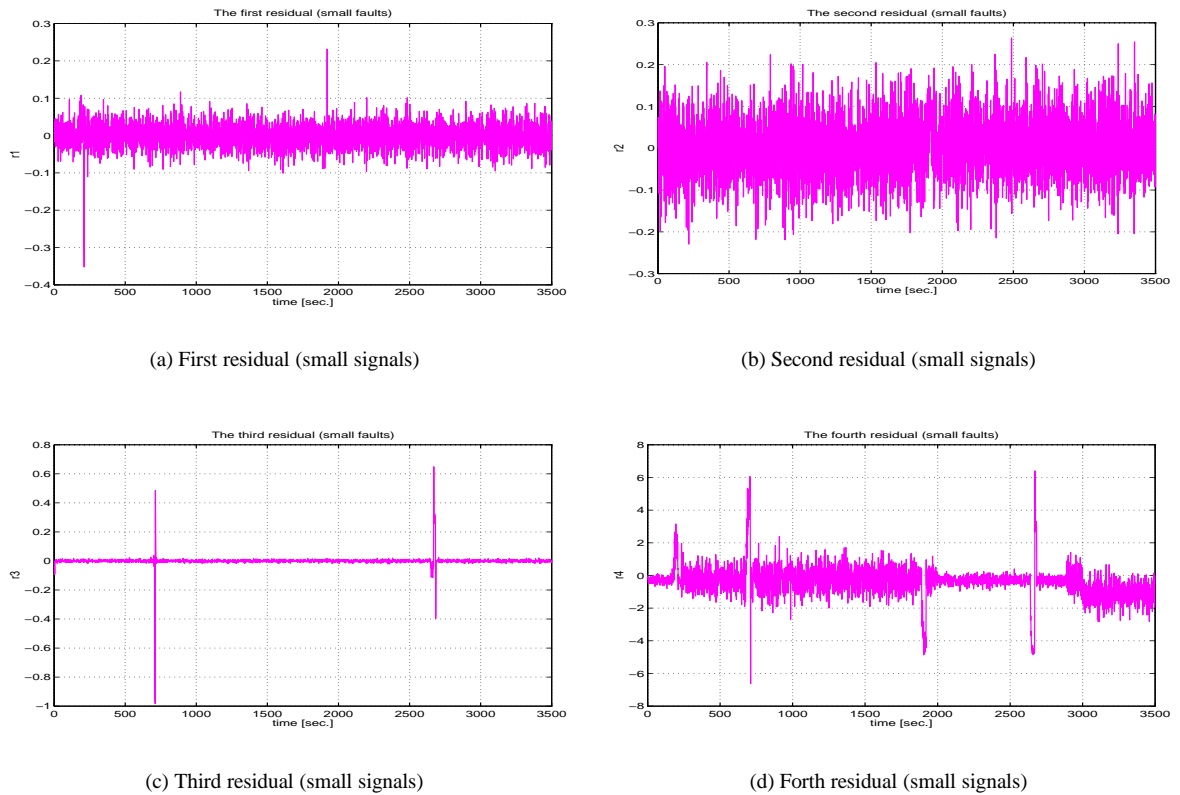(d) Forth residual (small signals)

Figure 4.1: The generated residuals when magnitude of the faults are small.

This approach consists of the monitoring of the innovations of a Kalman filter. In general, as the distribution of the innovations are given by a *conditional* distribution, the change detection function is calculated in the

following manner:

$$
\begin{aligned}
g_n &= \left(g_{n-1} + \log \frac{p_{\theta_1}(y_t | y_{t-1}, \cdots, y_{t-q})}{p_{\theta_0}(y_t | y_{t-1}, \cdots, y_{t-q})} \right) \\
&= g_{t-1} + s_t
\end{aligned}
\tag{4.18}
$$

where s_t is the cumulative sum value at time t and q denotes the number of past observations. Assume that following hypothesis are given (under the assumption that the innovations (residuals) are Gaussian distributed):

$$
\begin{aligned}
\mathbb{H}_0 &\quad : \quad y_t \in \mathcal{N}(\mu_0, \sigma_0^2) \quad : \quad \theta_0 = (\mu_0, \sigma_0^2) \\
\mathbb{H}_1 &\quad : \quad y_t \in \mathcal{N}(\mu_1, \sigma_1^2) \quad : \quad \theta_1 = (\mu_1, \sigma_1^2)
\end{aligned}
$$

In this case, the distribution of the signal/residual is not conditioned to its past values, i.e. $p_\theta(y_t | y_{t-1}, \cdots, y_{t-q}) = p_\theta(y_t)$. The cumulative sum value at time t, i.e. s_t is hence become:

$$
s_t = \ln\left(\frac{\sigma_0}{\sigma_1}\right) + \frac{1}{2\sigma_0^2}\left(y_t - \mu_0\right)^2 - \frac{1}{2\sigma_1^2}\left(y_t - \mu_1\right)^2
\tag{4.19}
$$

The decision function can recursively be calculated due to following function:

$$
g_t = \begin{cases}
g_{t-1} + s_t & \text{if} \quad g_t < \lambda \\
\lambda & \text{if} \quad g_t \geq \lambda \quad \text{alarm} \\
0 & \text{if} \quad g_t < 0
\end{cases}
\tag{4.20}
$$

4.2   Application

Both methods are implemented and tested on two sets of residuals. The first set, which is shown in figures 4.1 (a-d), includes the residuals generated for the system when the magnitude of the faults are small. The other set of the residuals, figures 4.2 (a-d), are generated in the case of faults with big magnitudes. The implementation results are shown in corresponding subsections. In order to compare the results and discussing their validity the time intervals for different fault events are shown in the following table. The total simulation time is 3500 sec..

Event	Fault type	Start time	End time
Angle measurement fault ($\Delta\theta_{high}$)	High	180 s	210 s
Leakage ($\Delta\dot{\theta}_{inc}$)	Incipient	800 s	1700 s
Angle measurement fault ($\Delta\theta_{low}$)	Low	1890 s	1920 s
Angular velocity measurement fault (Δn_{high})	High	680 s	710 s
Angular velocity measurement fault (Δn_{low})	Low	2640 s	2670 s
Gain fault (Δk_y)		3000 s	3500 s

4.2.1   Results using Unknown jump magnitude

This method, represented by equations (4.12), (4.13), (4.16), and (4.17), is applied to the first and the third residual since they are "ideal" residuals for this method. The results of FDI on the first residual are shown in table 4.1 .

Comment on the results in table 4.1:   Surveying the results, following statements can be given:

- it becomes obvious that the FDI algorithm in this case can not detect the small faults (rows 1 to 4). On the other hand, the FDI algorithm is quite efficient for detecting big faults (last three rows) and can also be used on-line by using equations (4.16), and (4.17). λ is been chosen to cover the variation of the residual in the interval $[-4 * \sigma^2 \quad 4 * \sigma^2]$ where σ^2 is the variance of the residual in non-faulty case. The interval is the $\% 99.99$ confidence interval for the residual in non-faulty case.
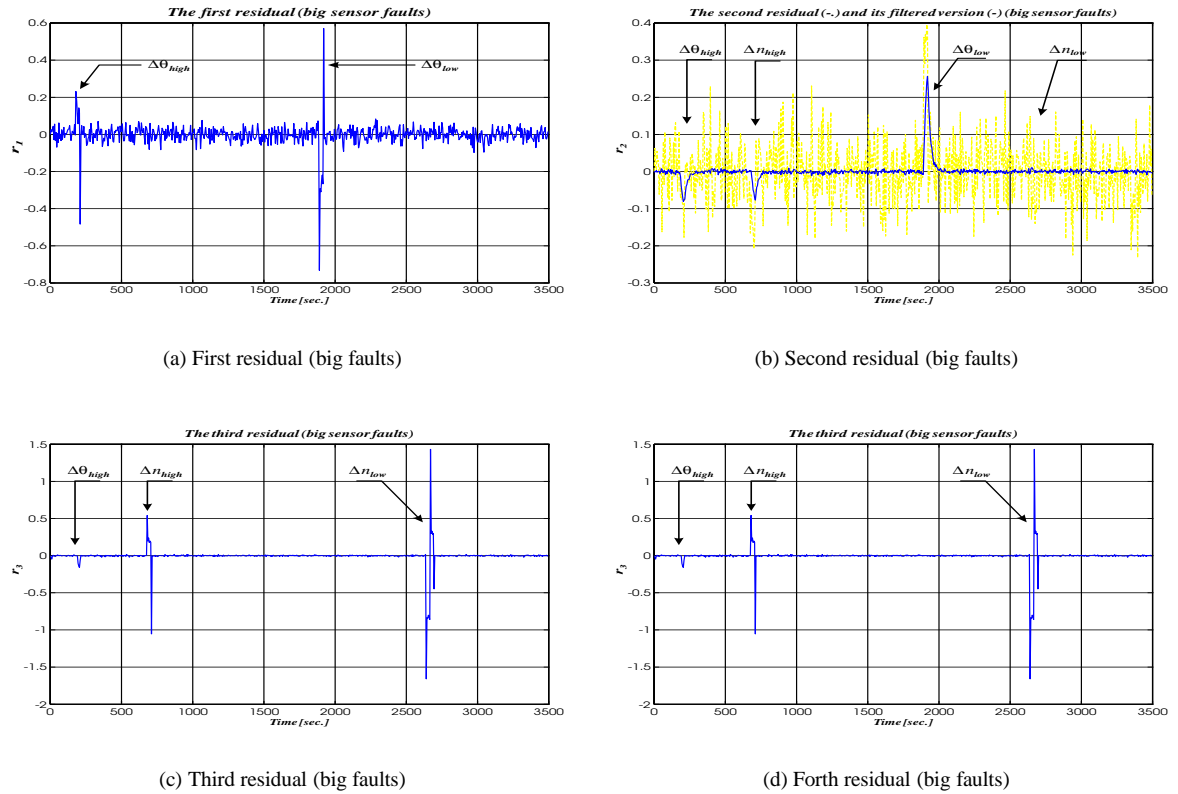
(a) First residual (big faults)

(b) Second residual (big faults)

(c) Third residual (big faults)

(d) Forth residual (big faults)

Figure 4.2: The generated residuals when magnitude of the faults are big.

Type of fault	Trial number	μ_0	ν	w	σ^2	λ	1. detection time	2. detection time	3. detection time
Small	1	0	-	20 s	-	0.4	211 s	-	1532 s
Small	2	0	-	50 s	-	0.4	211 s	-	1921 s
Small	3	0	0.1	-	-	0.4	211 s	-	1921 s
Small	4	0	0.01	-	-	0.26	189 s	1533 s	1892 s
Big	1	0	-	20 s	-	0.26	181 s	1433 s	1890 s
Big	2	0	-	100 s	-	0.26	181 s	1433 s	1890 s
Big	3	0	0.01	-	-	0.26	180 s	1533 s	1890 s

Table 4.1: Detection results on the first residual using equations (4.12), (4.13), (4.16), and (4.17)

- It should be noticed that the occurrence of the small faults in the residuals in the reality shows the case where the propeller pitch has the "right angle". In other words, the small fault situation indicates the cases where the propeller pitch is close to its maximum positive (or maximum negative) angle in no-fault situation and the sensor fault occurs in the same direction, i.e. sensor generates maximum (or minimum) signal in faulty situation. From operational and safety point of view, these situations are not considered to have a high severity level as they do not pose any danger to either the ship or the crew. It should also be noticed that, these faults are still detectable as their behavior on the propeller system resembles the one of the incipient fault.

- Detection results on the signals with big faults, shown in the last three rows of the table 4.1, are quite satisfactory, as they meet the fault detection requirements: both sensor faults in closed loop (1. and 3. detection) are within 2 sample times. It should be noticed, that the real detection times occur 1 sample later than the ones shown in the table. This is due to calculation of the first residual where one future sample of the measurement is been used. The detection time for the incipient fault, i.e.

Type of fault	Trial number	μ_0	ν	w	σ^2	λ	1. detection time	2. detection time	3. detection time
Small	1	0	-	$20\ s$	-	0.4	-	$709\ s$	$2664\ s$
Small	2	0	-	$50\ s$	-	0.4	-	$709\ s$	$2664\ s$
Small	3	0	0.05	-	-	0.4	-	$708\ s$	$2663\ s$
Small	4	0	0.01	-	-	0.053	-	$701\ s$	$2650\ s$
Big	1	0	0.01	-	-	0.053	$190\ s$	$679\ s$	$2640\ s$
Big	2	0	-	$100\ s$	-	0.1625	$190\ s$	$679\ s$	$2640\ s$

Table 4.2: Detection results on the third residual using equations (4.12), (4.13), (4.16), and (4.17)

2. detection time, does not meet the time requirement which is set to be less than 100 samplings time. However, as the incipient fault occurs very slowly, it take quite a number of samples to detect a change in a mean value of the residual. A way to detect this fault would be to lower the threshold λ. But it is already chosen to meet the criteria of missed detection (< 0.001) and hence can not be lowered further. The result suggests that the time requirement for detection of the incipient fault should be relaxed.

Figure (4.3) depicts the results of the FD algorithm using a window of the length 20 sampling times, i.e. $w = 20\ s$ applied on the first residual with imposed big faults.

The results of FD algorithm on the third residual are shown in table 4.2.

Comment on the results in table 4.2: Surveying the results, following statements can be given:

- The FD results show the same pattern as the results for the first residual: The FD algorithm is able to detect the small fault but on the cost of the time. However, it can be argued (as in the case of the first residual) that this case in not serious and by relaxing the detection time requirement the performance of the algorithm will be acceptable.

- For big faults the, results of the FD algorithm, shown in the last two rows, indicate that time requirements for fault detection in closed loop are met ($< 2s$). The last two column show that the sensor faults (with big magnitudes) on the shaft speed are detected within 1 sample.

- The FD algorithm also detects one fault which occurs in the propeller pitch angle control loop ,i.e. $\Delta\theta_{high}$, but does not detect $\Delta\theta_{low}$, which stems from the same source, namely, the pitch angle sensor measurement!. This fault is detected too late (11 s) later than the occurrence of the fault. If the applied FD algorithm on the first residual is reliable, then this fault can be automatically disregarded using a simple logic. The reason behind the FD algorithm's ability to detect $\Delta\theta_{high}$ while missing
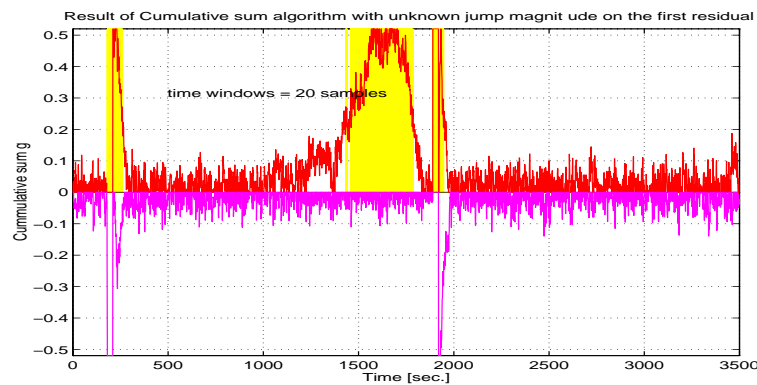

Figure 4.3: FD result on the first residual with big faults. $\lambda = 0.26$, $w = 20\ s$.

the fault $\Delta\theta_{low}$, is due to the physical limitations imposed on the Fuel system (constraint f_6). In this specific case, the imposed fault, $\Delta\theta_{low}$, will put a demand on the system to generate negative fuel. As this is not possible, the system will not react to this fault and hence the residual will not be affected. It should be emphasized that, this is a special case and on the other occasions, when it is possible, the system will react and we can detect the fault.

Figure 4.4 depicts the results of the FD algorithm using a window of the length 100 sampling times, i.e. $w = 100\ s$ applied on the third residual with imposed big faults. Notice that the variation of ν during the first 100 samples is quite large and causes false alarm generation. When this algorithm is implemented on the real plant, the results in the first samples (equal to the used time window) should therefore be disregarded. λ is been chosen to cover the variation of the residual in the interval $\begin{bmatrix} -5 * \sigma^2 & 5 * \sigma^2 \end{bmatrix}$ where σ^2 is the variance of the residual in non-faulty case.
Since the second and the forth residual involve tables and some parameter variations, the actual FD algorithm is not appropriate for application.

4.2.2 Assumptions for using GLR test (CUSUM test)

Following assumptions/simplifications are used for the implementation of equations (4.19) and (4.20).

First assumption: It is assumed that the variance of the signal is the same in both the no-fault situation as well as faulty one, i.e. $\sigma_0^2 = \sigma_1^2 = \sigma^2$. In this case the the equation (4.19) will become simplified to:

$$
\begin{aligned}
s_t &= \frac{1}{2\sigma^2}\left[\left(y_t - \mu_0\right)^2 - \left(y_t - \mu_1\right)^2\right] \quad (4.21)\\
&= \frac{1}{2\sigma^2}\left[2y_t\left(\mu_1 - \mu_0\right) - \left(\mu_1^2 - \mu_0^2\right)\right]\\
&= \frac{\nu}{\sigma^2}\left[y_t - \mu_0 - \frac{\nu}{2}\right]
\end{aligned}
$$

where ν is the magnitude of the jump. The recursive decision function given by equation (4.20) shall be used directly.

Second assumption: In most of the real cases the magnitude/mean value of the residual in the faulty case is not known. Furthermore, if the residual is sensitive to some slow parameter variations, its mean value will (slowly) change. This situation is quite actual for non-linear systems. In this case the mean values can be recursively calculated using similar equations as the equation (4.16) and (4.17). They are given in the
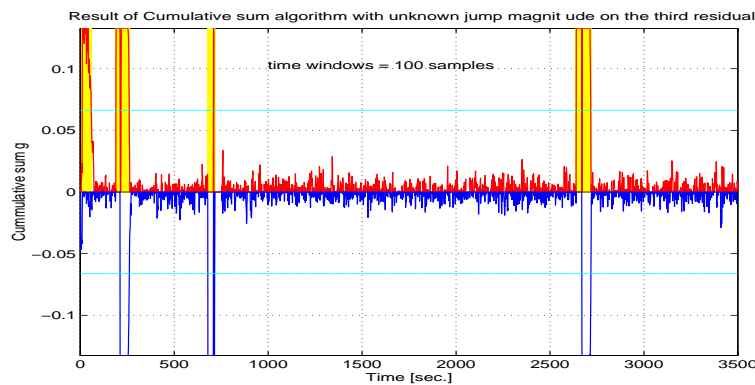

Figure 4.4: FD result on the third residual with big faults. $\lambda = 0.1625$, $w = 100\ s$.

following:

$$
\begin{aligned}
\hat{\mu}_i(t)\Big|_{w_i} &= \frac{1}{w_i} \sum_{k=t-w_i+1}^{t} y(k) \\
&= \hat{\mu}_i(t-1)\Big|_{w_i} + \frac{1}{w_i}\big(y(t) - y(t-w_i)\big) \quad\quad (4.22)
\end{aligned}
$$

where t denotes the actual time instant, $i \in \{0, 1\}$ indicates the situation (no-fault and fault). The size of the windows are given by w_0, and w_1 where $w_1 < w_0$. For the time instants less than the window's size, i.e. $t \in [1 \cdots w_i]$, following method can be used:

$$
\begin{aligned}
\hat{\mu}_i(t) &= \frac{1}{t} \sum_{k=1}^{t} y(k) \\
&= \hat{\mu}_i(t-1) + \frac{1}{t}\big(y(t) - \hat{\mu}_i(t-1)\big) \quad\quad (4.23)
\end{aligned}
$$

Third assumption: It is assumed that the variance is known and is constant. This will not hold for the case where there are slow changes in the mean value of the signal and the variance of the signal due to that[1]. This requires an on-line estimation of the variance as well, which is implemented in the following manner:

$$
\begin{aligned}
\hat{\sigma}^2(t)\Big|_{w_0} &= \frac{1}{w_0 - 1} \sum_{k=t-w_0+1}^{t} \big(y(k) - \hat{\mu}_0(k)\big)^2 \\
&= \hat{\sigma}^2(t-1)\Big|_{w_0} + \frac{1}{w_0 - 1}\left(\big(y(t) - \hat{\mu}_0(t)\big)^2 - \hat{\sigma}^2(t-1)\Big|_{w_0}\right) \quad (4.24)
\end{aligned}
$$

where the w_0 is the size of the window which was chosen for calculating $\hat{\mu}_0$. $t \geq w_0$ is the time instant and $\hat{\mu}_0$ is calculated by using equation (4.22). For time instants $t: \quad 1 < t < w_0$ following equation should be used:

$$
\begin{aligned}
\hat{\sigma}^2(t) &= \frac{1}{t-1} \sum_{k=1}^{t} \big(y(k) - \hat{\mu}_0(k)\big)^2 \\
&= \hat{\sigma}^2(t-1) + \frac{1}{t-1}\left(\big(y(t) - \hat{\mu}_0(t)\big)^2 - \hat{\sigma}^2(t-1)\right) \quad\quad (4.25)
\end{aligned}
$$

where $\hat{\mu}_0$ in this case is calculated using equation (4.23).

4.2.3 Results using GLR test (CUSUM test)

This method, represented by equations (4.18)-(4.20), is applied to the second and the fourth residual. There was a need for on-line calculation of mean values and the variance of the signals as they change due to some parameter variations. Hence, equations (4.21)-(4.25) are applied as well. The results of FD on the second residual are shown in table 4.3 .

Comment on the results in table 4.3: Surveying the results, following statements can be given:

- Observing the detection results for the small faults, it is noticed that the residual does not contains information enough to detect the faults in a proper way. In fact, the results are not truth-worthy at all: the FD algorithm detect the faults far too late to be used for anything. When the means are calculated on-line, (trial 2, Small), the FDI algorithm hasn't detect the real faults. It has, in fact, detected the transient of the system which has taken place due to removing the faults.

---

[1]It should be noticed that these are not related in principle. In other words, a slow change in the mean value does not mean that the variance will change as well. However, in most realistic cases this is the case and therefore can justify using an on-line estimation of the variance.

Type of signal	Trial nu.	μ_0	μ_1	ν	w_0	w_1	σ^2	λ	1. det. time	2. det. time	3. det. time	4. det. time
Small	1	0	±0.05	±0.05	-	-	0.0052	6	209 s	702 s	1904 s	2667 s
Small	2	-	-	-	50 s	48 s	0.0052	1	218 s	721 s	1926 s	2681 s
Big	1	0	±0.05	±0.05	-	-	0.0052	5	190 s	689 s	1892 s	-
Big	2	-	-	-	50 s	48 s	0.0052	1	212 s	718 s	1928 s	-

Table 4.3: Detection results on the second residual using equations (4.18)-(4.20), and (4.21)-(4.25)

Sig	No	μ_0	μ_1	ν	w_0	w_1	σ^2	λ	1. d. time	2. d. time	3. d. time	4. d. time	5. d. time
S	1	-	-	-	50	48	0.2864	1	213	681	1921	2670	3184
B	1	-	-	—	150	148	0.2864	6	201	714	2062	2670	3184
B	2	-	-	-	50	48	0.2864	7	194	709	1921	2670	—

Table 4.4: Detection results on the fourth residual using equations (4.18)-(4.20), and (4.21)-(4.25)

- The results with the big faults show the same pattern and are unreliable for detection. The final conclusion is that this residual is not useful for detection.
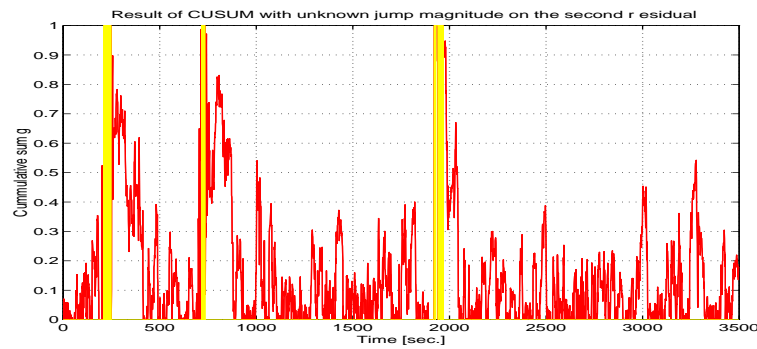

Figure 4.5: FD result on the second residual with big faults. $\lambda = 1$, $w_0 = 50\ s$, $w_1 = 48\ s$.

Figure 4.5 depicts the results of the CUSUM algorithm applied on the second residual with imposed big faults with estimated mean values (the last row in the table).

The results of CUSUM algorithm on the forth residual are shown in table 4.4 .

Comment on the results in table 4.4: Surveying the results, following statements can be given:

- When the faults are small the detector can detect all the occurred faults on the system, as the residual is detected by all of them. The same statement is also valid when the sensor faults are big and the size of the windows (w_0 and w_1) is big as well. This case is shown by the results in the second row.

- When the size of the windows decreases, the FD algorithm looses it's ability to detect the engine fault which is a multiplicative fault. Furthermore, since the mean value and the variance of this residual varies rapidly, the magnitude of the FD algorithm's result varies accordingly. This is depicted in figure 4.6.

- Non of the results in neither small fault case nor big fault case meet the time requirements. From this point of view, the detector's performance is not satisfactory.

Figure 4.6 depicts the results of the CUSUM algorithm applied on the fourth residual with imposed big faults with estimated mean values (the last row in the table).
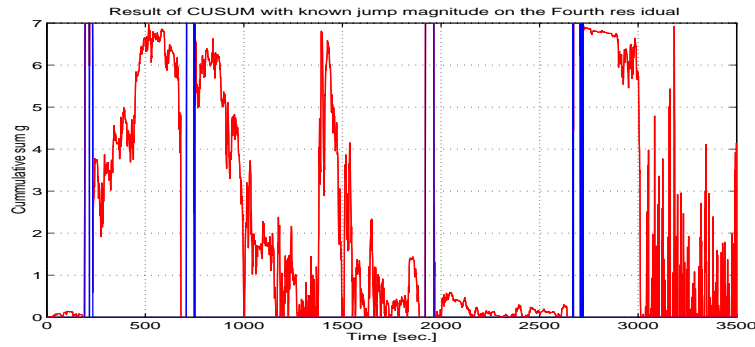
Figure 4.6: FD result on the forth residual with big faults. $\lambda = 7$, $w_0 = 50\ s$, $w_1 = 48\ s$, and $\sigma^2 = 0.2864$.

Sig	No	μ_0	μ_1	ν	w_0	w_1	σ^2	λ	1. d. time	2. d. time	3. d. time	4. d. time	5. d. time
B	2	-	-	-	50	48	0.2864	7	194	709	1921	2670	—
B	3	-	-	-	50	48	—	±0.15	181	680	1891	2640	3008
B	4	-	-	-	50	48	—	1	182	680	1891	2640	—

Table 4.5: Detection results on the fourth residual based on the variations on the estimated jump magnitude and variance

Detection based on the estimated jump magnitude and variance: As the mean value and the variance of this residual changes and since the CUSUM algorithm did not generated a satisfactory result, it was decided to base the test on the variation of the estimated jump, i.e. $\hat{\nu} = \hat{\mu_1} - \hat{\mu_0}$, and the calculated variance given by equations (4.24) and (4.25). The results are shown in table 4.5. The results based on the estimated jump magnitude shows that, this simple method can actually detect all sensor faults and the gain fault as well. The detection times shown in the table do not represent the actual time as the residual is obtained by numerical calculation. This introduces delay in the obtained residual. In the case of the fourth residual the time delay is 2 samplings time and should hence be added to the results in the table. The results are discussed in the following:

- Results of the test on the estimated jump magnitude (Trial 3, second row) shows that (big) sensor faults on the angular velocity measurements, i.e. Δn_{high} and Δn_{low}, are detected with two samples delay (due to calculation time). This test can, therefore, meet the requirements.
  Angle measurement faults, $\Delta\theta_{high}$ and $\Delta\theta_{low}$, are also detected. However, the detection time is 3 samples which does not meet the closed loop detection requirements.
  The gain fault Δk_y is detected after 10 sampling times. This, also, does not meet the requirement which is 5 samplings time. However, as this residual is the only one which contains detectable information about this fault, the time requirement has to be relaxed in order to use this FD method.

- Results of the test on the estimated variance (Trial 4, third row) show the same pattern for the unknown jump. The difference is that, it can not detect the gain fault.

The results of the detection by estimated jump magnitude and variance are depicted in figures 4.7a and 4.7b.

4.3   Discussion

The results of the detection methods applied on the different residuals are summarized in table 4.6. The $\div$ indicates that the corresponding fault has been detected, but the result is useless since it does not meet the time requirements (too late fault detection). As it is shown in table, it is possible to detect and isolate the defined faults, although one of the residuals, namely the second residual, does not provide any reliable result. The fault isolation, however, should be performed in an intelligent manner:
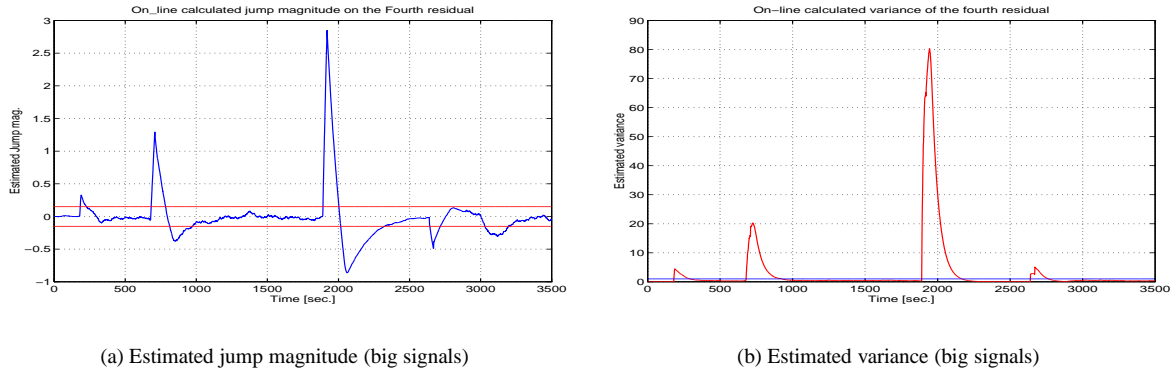
(a) Estimated jump magnitude (big signals)

(b) Estimated variance (big signals)

Figure 4.7: The generated residuals when magnitude of the faults are small.

The fault	$\Delta\theta_{high}$	$\Delta\dot{\theta}_{inc}$	$\Delta\theta_{low}$	Δn_{high}	Δn_{low}	Δk_y
Occurrence time	180	800	1890	680	2640	3000
$resid_1$	182	1434	1891			
$resid_2$	$\div$		$\div$	$\div$		
$resid_3$	$\div$		$\div$	680	2641	
$resid_4$	$\div$		$\div$	$\div$	$\div$	
$\hat{\nu}_{r_4}$	183		1893	682	2642	3008
$\hat{\sigma}^2_{r_4}$	184		1893	682	2642	

Table 4.6: The summary of the detection results on the 4 residuals, The last two rows are test on the 4th residual based on the estimated jump magnitude and the variance.

- *Detection of the faults in propeller pitch control loop:* As it is shown in the first row of the table, the sensor faults ,i.e. Δ_{high} and $\Delta\theta_{low}$ can be detected while meeting the time requirement. On the other hand, since the same detector can also detect the incipient fault $\Delta\dot{\theta}_{inc}$, there will be need for additional information. For that purpose, the results of detection on the $\hat{\nu}_{r_4}$, and $\hat{\sigma}^2_{r_4}$ can be used. As they do not meet the time requirement (they occur 1 sample later than allowed) following strategy should be used:

  The sensor faults have the highest severity level [referencesafeprocess] so they should be handled immediately. When the detector on the first residual detects a fault, it should be handled, under all circumstances, as if there is a sensor fault and remedy action should be taken. If the fault is really a sensor fault, then it can be verified by checking $\hat{\nu}_{r_4}$ and $\hat{\sigma}^2_{r_4}$. Otherwise, the remedy action should be terminated/modified to handle the resulting incipient fault.

- *Detection of the sensor fault in shaft speed control system:* This fault can be very effectively detected using $resid_3$, $\hat{\nu}_{r_4}$, and/or $\hat{\sigma}^2_{r_4}$ by following rule:

$$\textbf{if } resid_3 == ON \ \wedge \ \hat{\nu}_{r_4} == ON$$
$$\textbf{Then } \text{Shaft sensor has failed.}$$

  Other combinations are also possible.

- *Gain fault*: The gain fault can be easily detected by using the following logic:

$$\textbf{if } \hat{\nu}_{r_4} == ON \ \wedge \ resid_1 == OFF \ \wedge \ resid_3 == OFF \ \wedge \ \hat{\sigma}^2_{r_4} == OFF$$
$$\textbf{Then } \text{Gain fault has occurred.}$$

It should be noticed that, the time requirements for the incipient fault and gain fault can not be held and there is a need for the relaxation of them. Furthermore, the detection time of the incipient fault, which is due to leakage in the pitch control system, is quite dependent on the magnitude of the leakage.

4.3.1 Establishing a relation for finding optimal windows for $\hat{\nu}_{r_4}$

During the performed detection test using $\hat{\nu}_{r_4}$, it became interesting to establish a way of deciding the size of windows for $\hat{\nu}_{r_4}$. Following approach was used:

The calculation of the unknown jump in residual 4 is based on the estimation of two mean values, μ_0 and μ_1. The former one is estimated on a window with size w_0 and another window with size w_1 is used to estimate the later one. They are iterated in the following equation:

$$\hat{\mu}_0(t)\Big|_{w_0} \quad = \quad \frac{1}{w_0} \sum_{k=t-w_0+1}^{t} y(k) \tag{4.26}$$

$$\hat{\mu}_1(t)\Big|_{w_1} \quad = \quad \frac{1}{w_1} \sum_{k=t-w_1+1}^{t} y(k) \tag{4.27}$$

where $w_0 > w_1$. The difference between the size of the windows are denoted by d, i.e. $d = w_0 - w_1$. $\hat{\mu}_0$ can be written as:

$$\hat{\mu}_0(t) = \frac{1}{w_0} \left(\sum_{k=t-w_0+1}^{t-w_1} y(k) + w_1 \hat{\mu}_1(t) \right)$$

Using the definition of unknown jump, we get:

$$\hat{\nu}(t) \quad = \quad \hat{\mu}_1(t) - \hat{\mu}_0(t)$$
$$= \quad \frac{1}{w_1} \left(\sum_{k=t-w_0+1}^{t-w_1} (y(k) - \hat{\mu}_0) \right) \tag{4.28}$$

where where the variations of $\hat{\mu}_0$ are assumed to be negligible. Equation (4.28) shows that the unknown jump is actually estimated over a small window with the size d. The equation can hence be written as:

$$\hat{\nu} = \frac{1}{w_1} \left(\sum_{k=1}^{d} (y(k) + \hat{\mu}_0) \right) \tag{4.29}$$

In no-fault condition the following statement is valid:

$$\hat{\mu}_0 \simeq \hat{\mu}_1 \quad \Longleftrightarrow \quad \hat{\nu} \simeq 0 < \epsilon$$

where ϵ is a small value. There are actually two parameters to decide. The one is d and the other one is w_1 which is the size of the shortest window. A relation can be established between these two variables by choosing the value of ϵ and defining the following:
Assume ϵ and w_1 are given. Choose d in order to fulfill the following requirements:

$$p\left\{ -\epsilon < \frac{1}{w_1} \left(\sum_{k=1}^{d} (y(k) - \hat{\mu}_0) \right) < \epsilon \right\} > P \tag{4.30}$$

where $y(k) \in \mathcal{N}(\mu_0, \sigma^2)$. The value of P should be chosen so that it fulfills the requirement for false detection. Typical values for P are $0.99, \ 0.995$, or 0.999. Since each observation is independent on the previous observations, equation (4.30) can be written as (under the white noise assumption):

$$p\left\{ \frac{-\epsilon w_1}{d} < y(k) - \hat{\mu}_0) < \frac{\epsilon w_1}{d} \right\} > P \tag{4.31}$$

Since we assumed that $y(k) \in \mathcal{N}(\mu_0, \sigma^2)$, then $x = (y(k) - \mu_0) \in \mathcal{N}(0, \sigma^2)$. The value of the σ should be known in advance. under this assumption we can obtain:

$$p\left\{ -b < x < b \right\} = 2F(b) - 1 = P \tag{4.32}$$

where

$$F(b) = p\{x < b\}, \qquad \text{and} \qquad b = \frac{\epsilon w_1}{d\sigma}$$

$F(b)$ can be calculated from equation (4.32), and b can thus be found by looking up in given statistical tables. The relation then will be

$$b = \frac{\epsilon w_1}{d\sigma} \quad \text{or} \quad w_0 = \left(1 + \frac{b\sigma}{\epsilon}\right)d \tag{4.33}$$

Where this relation can be used: This relation can be used for the residuals with slow mean values and constant variance, which in other words means that they should closely resemble normal distributed noises. This was not the case for the forth residual (both mean value and variance varied quite a lot). Furthermore, the residual was affected by several colored noise stemming from different measurement sources in the closed-loop control system and the uncertainties from the table. However, it did provide some heuristic guidelines for choosing the windows' size. Additional tests (with different residuals) are needed to verify the obtained relation.

Chapter 5

Conclusion

The structural approach methodology has been applied on the ship control benchmark. The essential part of the theory which is used in this approach is reviewed in the report. The structural approach employs well-defined di-graph theory, which makes it possible to develop a software tool that can fully support design engineers. One of the main advantages of this approach is that with even course information available about the system, the monitorable part (the over-determined part) of the system can be found. It hence enables design engineers to use the structural approach during at any step of the design phase while using the available information. However, detailed information is needed to compute the residuals (ARRs).

Using this approach it was possible to generate 4 analytical redundancy relations (residuals). Two different fault detection algorithms were used to detect the defined faults. The algorithms are: *change detection in jump* algorithm and *generalized likelihood ratio test (CUSUM test)* algorithm. The first algorithm which originally is an off-line algorithm was modified slightly for on-line fault detection purposes. This algorithm showed to be quite efficient in detecting faults (both incipient and sensor faults) under the condition that the mean value (and variance) of the residual does not change (in no-fault case). The second algorithm was used on residuals with slight variation in their mean value and variance. It was necessary to calculate the mean value and variance on-line in order to obtain reliable results.

The results of implementing the structural approach on the ship benchmark and the following FDI methods can be summarized in the following:

- **Structural approach**: showed to be a powerful approach that can be used during all the steps of the design phase. The structural approach was used to determine the parts of the system on which four ARRs could be generated and also was used to obtain the calculation sequences of the ARRs (residuals). Two of the obtained four residual expressions contained tables of real data. Because of these tables it was not possible to calculate the sensitivity of these two residuals w.r.t. different faults. In such cases only simulation tests can be used to evaluate the sensitivity.
  The third residual is an interesting residual. The evaluation form of this residual (showed in table 3.1) indicates that this residual is only sensitive to sensor fault Δn_m. However, the simulation showed that this residual is also sensitive to sensor fault $\Delta\theta$. The cause of this phenomenon showed to be due to cross-coupling effect: there is a coupling between the pitch angle control loop and the shaft speed control loop. An unpredicted change in the sensor output can be caused either by occurrence of a fault in it or by sudden change in the pitch angle control loop.

  This example emphasizes the need for having a deep knowledge about the system and to understand its behavior during the FDI design phase in order to acquire relaible results.

- **Time requirements**: There are two possible ways of dealing with this issue:

  1. Developing of advanced (and possibly specialized) FD algorithms to detect the specific faults in the system with the aim that time requirements are met. Since these algorithms should also

be robust to uncertainties in the real system, it becomes quite time consuming (and expensive) to achieve this objective.

2. The other way is to use more simple and robust methods which are known and then relax the time requirements. This leads to another problem that is to achieve bump-less transition from faulty to non-faulty state in the close loop. Therefore, a controller adjustment (or eventually controller redesign) will be needed to avoid the new problem. .

As there is no requirement on the bump-less transition for the ship benchmark, an slight relaxation of time requirements will be necessary in order to isolate the faults properly

- **Fault isolation**: A complete design of the logic aimed at isolating the faults normally requires a set of models representing the system's behavior under different operational situations. The isolation logic presented in this report covers only the normal operation of the ship (move forward, moderate acceleration).

The FDI results for the defined model of the benchmark are satisfactory and meet most of the time requirements.

Alternative methods for generating residuals (f.x. non-linear observers) and performing FDI should be explored in order to compare and evaluate the obtained results.

Bibliography

[BN94]   M. Basseville and I. Nikiforov. *Statistical Change Detection*. Prentice Hall, 1994.

[CLCS94] J.Ph. Cassar, R.G. Litwak, V. Cocquempot, and M. Staroswiecki. Approche structurelle de la conception de systèmes de surveillance pour des procédé industriels complexes. *Revue Européenne de Diagnostic et Sûreté de fonctionnement*, 1994. Vol.4, no. 2, pp. 179-202.

[DS03]   P. Declerck and M. Staroswiecki. Characterization of the canonical components of a structural graph for fault detection in large scale industrial plants. In *Proceedings of ECC'91*, Grenoble, France, July 1991, pp. 298-303.

[IZB97]   R. Izadi-Zamanabadi and M. Blanke. A ship propulsion system as a benchmark for fault-tolerant control. *IFAC Safeprocess 97, Hull, England*, August 1997. pp 1074-1082.

[SD27]   M. Staroswiecki and P. Declerck. Analytical redundancy in non-linear interconnected systems by means of structural analysis. In *IFAC-AIPAC'89*, Nancy, July 1989, vol.II, pp. 23-27.