Universitat de Girona

TIME MISALIGNMENTS IN FAULT DETECTION AND DIAGNOSIS

David Alejandro LLANOS RODRÍGUEZ

ISBN: 978-84-691-8900-9
Dipòsit legal: GI-1593-2008

Universitat de Girona
Departament d'Electrònica, Informàtica i Automàtica

Time Misalignments in Fault Detection and Diagnosis

by

David Alejandro Llanos Rodríguez

Advisors

Dr. Joaquim Meléndez, Dr. Joan Colomer and Dr. Marcel Staroswiecki

DOCTORAL THESIS
Girona, Spain
October, 2008

UNIVERSITAT DE GIRONA
DEPARTAMENT D'ELECTRÒNICA, INFORMÀTICA I AUTOMÀTICA

TIME MISALIGNMENTS IN FAULT DETECTION AND DIAGNOSIS

by

David Alejandro Llanos Rodríguez

A dissertation presented to the Universitat de Girona in partial fulfillment of the requirements of the degree of DOCTOR OF PHILOSOPHY

Advisors

Dr. Joaquim Meléndez

Dr. Joan Colomer

Dr. Marcel Staroswiecki

Girona-Spain, October, 2008

Universitat de Girona
Departament d'Electrònica, Informàtica i Automàtica

Abstract

Time Misalignments in Fault Detection and Diagnosis

by David Alejandro Llanos Rodríguez

Advisors: Dr. Joaquim Meléndez, Dr. Joan Colomer and Dr. Marcel Staroswiecki

October, 2008
Girona, Spain

The design of control, estimation or diagnosis algorithms most often assumes that all available process variables represent the system state at the same instant of time. However, this is never true, because of the time misalignments. Time misalignment is the unmatching of two signals due to a distortion in the time axis of one or both signals. Potential sources of time misalignments are: different time response among sensors, data communication problems, analog to digital conversion, sensor location, and so on. Fault Detection and Diagnosis (FDD) deals with the timely detection, diagnosis and correction of abnormal conditions of faults in a process. The methodology used in FDD is clearly dependent on the process and the sort of available information and it is divided in two categories: model-based techniques and non-model based techniques. This doctoral dissertation deals with the study of time misalignments effects when performing FDD. Our attention is focused on the analysis and design of FDD systems in case of data communication problems, such as data dropout and time delays due to data transmission. Techniques based on dynamic programming and optimisation are proposed to deal with these problems. Numerical validation of the proposed methods is performed on different dynamic systems: a control position for a DC motor, a the laboratory plant and an electrical system problem known as voltage sag.

Acknowledgments

I have had the privilege of working with three co-advisers. Dr. Joaquim Meléndez and Dr. Joan Colomer who accepted me in eXiT (enginyeria de Control i Sistemes inTel·ligents), the research group that they have nurtured at the Univesitat de Girona. They both shared with me their expertise in the field of Fault Detection and Diagnosis for which I would always be grateful. Their guidance during the structuring and development phases of this thesis is also highly appreciated. Dr. Marcel Staroswiecki, from the University of Lille, offered me the opportunity of spending three months at the LAIL research group of École Polytechnique Universitaire de Lille. The mathematical approach used in this thesis was developed by Dr. Staroswiecki as well as the definition of the problem statement. I appreciate in particular, Dr. Staroswiecki encouragement and the high standard that he demands as well as the promptness of his replies while editing this manuscript. To all of them I am forever grateful as their mentoring and input to this work through our discussions have contributed in shaping the way I approach scientific problems.

To the Universitat de Girona for the financial support through a research grant (BR01/01). Additionally, for the economical support during my 3 months as a visitor researcher at the LAIL research group of École Polytechnique Universitaire de Lille, France.

This work has been partially supported by the projects "Desarrollo de un sistema de control y supervisión aplicado a un reactor secuencial por cargas para la eliminación de materia orgánica, nitrógeno y fósforo" (DPI2002-04579-C02-01) and DPI SECSE - "Supervisión Experta de la Calidad de Servicio Eléctrico" (DPI2001-2198) within the CICYT program from the Spanish government and FEDER funds, which are hereby gratefully acknowledged.

To the Universitat de Girona and to the company Sociedad de Explotación de Aguas Residuales S.A. -SEARSA-, for supporting me through the Agreement of Educational Cooperation (705/2005-06).

To my wife Rosa who with love and patience encouraged me during crucial moments of my thesis. And overall, thanks for the so beautiful gift that you have given me, our daughter Paula, who is my Sun of every day.

To my parents Sebastián and Mariela who with their example, efforts and love make the dreams of their children and grandchildren into reality. I would also like to express my gratitude to my siblings Flavio, Lyra, Ella, Rocío and Jaime and my nieces Silvia, Isabela and Alejandra. They all from distant places, but with the same unconditionality and encouragement, were motivating me during this period.

To my family in Girona: Jose Maria, Paquita, Jordi, Maria and Julia, who have given me great moments and make me feel like at home.

To the members of eXiT research group. Particular thanks to Francisco Gamero for sharing me the Time Warping Toolbox and for helping me with the development of the $DTW_{on-line}$ algorithm. I would also like to thank Juan José Mora and Magda Ruíz for their collaboration in the field of Power Quality. I also want to mention a student who contributed whit his work to the elaboration of some examples described in this work, this is the case of Xavier Berjaga. To all of them many thanks for their friendship and support during this time.

It is difficult to remember and to include every person who directly or indirectly has collaborated in the development of this thesis. I offer my sincere apologies but I will ever thank them.

Agradecimientos

He tenido el privilegio de haber trabajado con tres tutores. Dr. Joaquim Meléndez y Dr. Joan Colomer los cuales me aceptaron en el grupo eXiT (ingeniería de Control y Sistemas inTeligentes), grupo de investigación que dirigen en la Univesitat de Girona. Ambos han compartido conmigo sus conocimientos en el campo de Detección y Diagnóstico de Fallos, por lo cual les estaré siempre agradecido. Su guía durante la estructuración y el desarrollo de las diferentes fases de esta tesis es muy apreciada. Dr. Marcel Staroswiecki, de l'Université de Lille, me ofreció la oportunidad de estar tres meses en el grupo de investigación LAIL de l'École Polytechnique Universitaire de Lille, Francia. El procedimiento matemático y la definición del problema han sido desarrollados por el Dr. Staroswiecki. Aprecio en particular los ánimos del Dr. Staroswiecki, el alto nivel y la inmediatez de sus respuestas durante la edición de este manuscrito. A todos ellos les estaré siempre agradecido por sus aportaciones en este trabajo que han contribuido a mi formación como investigador.

A la Universitat de Girona por el soporte financiero con la beca de investigación (BR01/01). Igualmente, por el soporte económico durante mis tres meses como investigador visitante en el grupo de investigación LAIL de l'École Polytechnique Universitaire de Lille.

Este trabajo ha sido parcialmente apoyado por los proyectos "Desarrollo de un sistema de control y supervisión aplicado a un reactor secuencial por cargas para la eliminación de materia orgánica, nitrógeno y fósforo" (DPI2002-04579-C02-01) y DPI SECSE - "Supervisión Experta de la Calidad de Servicio Eléctrico" (DPI2001-2198) dentro del CICYT programa del gobierno Español y el fondo FEDER.

A la Universitat de Girona y a la empresa Sociedad de Explotación de Aguas Residuales S.A. -SEARSA-, por su apoyo a través del Convenio de Cooperación Educativa núm. 705/2005-06.

A mi esposa Rosa quien con su amor y paciencia me ha animado en los momentos cruciales de mi tesis. Además, gracias por el maravilloso regalo que me ha dado, nuestra hija Paula, que es mi Solecito de cada día.

A mi padres Sebastián y Mariela quien con su ejemplo, esfuerzo y amor han hecho realidad los sueños de sus hijos y nietos. También quiero expresar mi gratitud hacia mis hermanos Flavio, Lyra, Ella, Rocío y Jaime y mis sobrinas Silvia, Isabela y Alejandra.

Todos ellos desde sitios distantes, pero con la misma incondicionalidad y apoyo, me han motivado durante este período.

A mi familia en Girona: Jose Maria, Paquita, Jordi, Maria y Júlia, con los cuales he compartido buenos momentos y me hacen sentir como en casa.

A los miembros del grupo de investigación eXiT . Particularmente gracias a Francisco Gamero por compartir conmigo el "Time Warping Toolbox" y por ayudarme con el desarrollo del algoritmo $DTW_{on-line}$. Quiero también dar las gracias a Juan José Mora y Magda Ruíz por su colaboración en el campo de Sistemas de Distribución de Energía Eléctrica. Igualmente quiero mencionar a un estudiante que ha contribuido con su trabajo en la elaboración de algunos ejemplos descritos en este trabajo, es el caso de Xavier Berjaga. A todos ellos muchas gracias por su amistad y apoyo durante todo este tiempo.

Es difícil recordar e incluir a todas las personas que directamente o indirectamente han colaborado en el desarrollo de esta tesis. Ofrezco mis sinceras disculpas y al mismo tiempo les doy las gracias.

Agraïments

He tingut el privilegi d'haver treballat amb tres tutors. Dr. Joaquim Meléndez i Dr. Joan Colomer els quals em van acceptar al grup eXiT (enginyeria de Control i Sistemes inTel·ligents), grup de recerca que dirigeixen a la Univesitat de Girona. Ambdós han compartit amb mi els seus coneixements en el camp de Detecció i Diagnosis de Falles, pel qual els estaré sempre agraït. La seva guia durant l'estructuració i el desenvolupament de les fases d'aquesta tesi és molt apreciada. Dr. Marcel Staroswiecki, de l'Université de Lille, em va oferir l'oportunitat d'estar tres mesos al grup de recerca LAIL de l'École Polytechnique Universitaire de Lille, França. El procediment matemàtic i la definició del problema ha estat desenvolupat pel Dr. Staroswiecki. Aprecio en particular l'encoratjament del Dr. Staroswiecki, l'alt nivell i la immediatesa de les seves respostes durant l'edició d'aquest manuscrit. A tots ells els estaré sempre agraït per les seves aportacions en aquest treball que han contribuït a la meva formació com a investigador.

A la Universitat de Girona pel suport financer amb la beca de recerca (BR01/01). Igualment, pel suport econòmic durant els meus tres mesos com a investigador visitant al grup de recerca LAIL de l'École Polytechnique Universitaire de Lille.

Aquest treball ha estat parcialment recolzat pels projectes "Desarrollo de un sistema de control y supervisión aplicado a un reactor secuencial por cargas para la eliminación de materia orgánica, nitrógeno y fósforo" (DPI2002-04579-C02-01) i DPI SECSE - "Supervisión Experta de la Calidad de Servicio Eléctrico" (DPI2001-2198) dins de CICYT programa del govern Espanyol i el fons FEDER.

A la Universitat de Girona i a l'empresa Sociedad de Explotación de Aguas Residuales S.A. -SEARSA-, pel seu suport a través del Conveni de Cooperació Educativa núm. 705/2005-06.

A la meva dona Rosa qui amb el seu amor i paciència m'ha encoratjat en els moments crucials de la meva tesi. I a més, gràcies pel meravellós regal que m'ha donat, la nostra filla Paula, que és el meu Solet de cada dia.

Als meus pares Sebastián i Mariela qui amb el seu exemple, esforç i amor han fet realitat els somnis dels seus fills i néts. També vull expressar la meva gratitud cap als meus germans Flavio, Lyra, Ella, Rocío i Jaime i les meves nebodes Silvia, Isabela i Alejandra. Tots ells des de llocs distants, però amb la mateixa incondicionalitat i encoratjament, m'han motivat durant aquest període.

A la meva família a Girona: Jose Maria, Paquita, Jordi, Maria i Júlia, amb els quals he compartit bons moments i em fan sentir com a casa.

Als membres del grup de recerca eXiT . Particularment gràcies a Francisco Gamero per compartir amb mi el "Time Warping Toolbox" i per ajudar-me amb el desenvolupament de l'algoritme $DTW_{on-line}$. Vull també donar les gràcies a Juan José Mora i Magda Ruíz per la seva col·laboració en el camp de Sistemes de Distribució d'Energia Elèctrica. Igualment vull mencionar a un estudiant que ha contribuït amb el seu treball en l'elaboració d'alguns exemples descrits en aquest treball, és el cas de Xavier Berjaga. A tots ells moltes gràcies per la seva amistat i suport durant tot aquest temps.

És difícil recordar i incloure a totes les persones que directament o indirectament han col·laborat en el desenvolupament d'aquesta tesi. Ofereixo les meves sinceres disculpes i al mateix temps els dono les gràcies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Due to the growing complexity and spatial distribution of automated systems, communication networks have become the backbone of most control architecture. As systems are required to be more scalable and flexible, they have additional sensors, actuators and controllers, often referred to as field (intelligent) devices (Lee et al., 2001, Staroswiecki, 2005). Networked Control Systems (NCS) result from connecting these system components via a communication network such as CAN (Controller Area Network), PROFIBUS or Ethernet. Control over data networks has many advantages compared with traditional control systems, such lower cost, greatly reduced wiring, weight and power, simpler installation and maintenance and higher reliability. However, the design of control, estimation or diagnosis algorithms most often are affected by time misalignments. Time misalignment is the unmatching of two signals due to a distortion (expansion or compression) in the time axis of one or both signals. Time misalignment can be produced by: different time response among sensors, analog to digital conversion, sensor location, data dropout, limited bandwidth, time delay due to data transmission, asynchronous clock among network nodes and other peculiarities of networks that could degrade the performances of the closed-loop systems and even destabilize them. The aforementioned problems have been intensively studied by the control community in the last several years (Zhang et al., 2001; Walsh and Hong, 2001; Walsh et al., 2002; Zhivoglydov and Middleton, 2003; Savkin and Petersen, 2003; Matveev and Savkin, 2003; Lian et al., 2003; Hu and Zhu, 2003; Ma and Fang, 2005; Li and Fang, 2006), including: analysis of impact of network on control performance, design of control algorithm taking into account the above factors and proposal of new network protocol suitable for control. Nevertheless, only a few studies of the impact of the communication network on the diagnosis of continuous systems have been recently published (Zhang et al., 2004; Ding and Zhang, 2005; Fang et al., 2006). The challenging problem that has motivated this thesis is the time misalignments effects when performing Fault Detection and Diagnosis (FDD).

1.2 Objectives

In this thesis the problem of time misalignments when performing FDD is investigated. The main objective of the thesis is to design techniques aiming at the minimisation of false alarms caused by transmission delay and data dropout without increasing the number of missed detection. The techniques rely on the explicit modelling of the communication network. Techniques based on dynamic programming and optimisation are proposed to deal with these problems. Numerical validation of the proposed methods is performed on different dynamic systems: a control position for a DC motor, a the laboratory plant and an electrical system problem known as voltage sag.

1.3 Thesis Outline

The contents of the thesis are as follows:

Chapter 2: Fault Detection and Diagnosis

This chapter introduces the terminology used in the field of fault detection and diagnosis. An overview of various diagnostic methods from different perspectives is also provided. The general formulation that is used in next chapters and a discussion about problems in fault detection and diagnosis are also presented. More emphasis has been put on time misalignments which is the challenging problem that has motivated this thesis.

Chapter 3: Time Misalignments in Supervisory Systems

The concept of time misalignments is defined in this chapter and potential sources of time misalignments are presented. Different models of network delays are given. The problem formulation of time misalignment and communication delays from model-based and non model-based fault diagnosis perspectives are stated. The associated work done in both directions is also presented.

Chapter 4: Delay Estimation for Residual Computation

In this chapter, a technique aiming to the minimization of the false alarms caused by transmission delays without increasing the number of missed detection is proposed. The technique relies on the explicit modelling of communication delays, and their most likely estimation. Application on a control position for a DC motor is shown.

Chapter 5: Reduction of False Alarms in Fault Detection of Networked Control Systems with Data Dropout

In this chapter, a technique aiming at the reduction of the false alarms caused by data dropout without increasing the number of missed detection is proposed. Illustrative example on a laboratory plant is given.

Chapter 6: Dynamic Time Warping for Residual computation

This chapter proposes the use of Dynamic Time Warping (DTW) to reduce the effects of time misalignment when residual computation is performed. The first section, Section 6.2, summarises the DTW algorithm. In Section 6.3, a modification of DTW in order to be applied on-line is explained. In Section 6.4 the use of DTW for improving the residual computation in the presence of time misalignment is formulated. Finally, Section 6.5 presents the applications in a laboratory plant of the University of Girona eXiT group.

Chapter 7: Time Misalignment Reduction in Symptom Based Diagnosis

This chapter proposes the use of Dynamic Time Warping (DTW) for reducing the effects of time misalignment when Case Based Reasoning (a symptom based diagnosis) is performed. DTW is used as a similarity criteria to implement the retrieval task. An electrical system problem, known as voltage sag, has been used to test the proposed method.

Chapter 8: Conclusions

In the last chapter conclusions are presented. Extensions and open problems are discussed.

Appendices

The thesis finishes with four appendices:

- A: Design of the analytical redundancy relation of a control position for a DC motor.
- B: Design of the analytical redundancy relations of the laboratory plant.
- C: Design of the analytical redundancy relations of the three tanks system.
- D: Temporal attributes of the voltage sags.

Chapter 2

Fault Detection and Diagnosis

2.1 Introduction

Fault Detection and Diagnosis (FDD) deals with the timely detection, diagnosis and correction of abnormal conditions of faults in a process. Early detection and diagnosis can help to avoid abnormal events progression and to reduce production loss. There is a wide literature documentation on process fault diagnosis ranging from analytical methods to artificial intelligence and statistical approaches. From a modelling perspective, there are methods that require accurate process models, semi-quantitative models or qualitative models. However, there are methods that do not assume any form of model information and rely only on historic process data. Therefore, this chapter is mainly devoted to introduce the terminology used in the field of fault detection and diagnosis, to provide an overview of various diagnostic methods from different perspectives, to introduce the general formulation that is used in next chapters and a discussion about drawbacks in fault detection and diagnosis are also presented, doing more emphasis in time misalignments which is the challenging problem that has motivated this thesis.

This chapter is organized as follows: Section 2.2 presents some terminologies and definitions that are used in the FDD field. Model-based techniques are gathered in Section 2.3. In contrast, Section 2.4 explains the non model-based techniques. Problems in FDD are mentioned in Section 2.5. Finally some concluding remarks are given in Section 2.6.

2.2 Terminology and definitions

The terminology used in the field of fault detection and diagnosis is not unique. Consequently, the Safeprocess Technical Committee of IFAC (the International Federation of Automatic Control) compiled a list of suggested definitions (Isermann and Ballé, 1997) which is in accordance with the terminology used this thesis.

About the states and the signals

- *Fault*: Unaccepted deviation of at least one characteristic property or parameter of a system from its acceptable/usual/standard condition.

- *Failure*: inability of a system or a component to accomplish its function.

- *False alarm*: is an indication of a fault, when in actuality a fault has not occurred.

- *Error*: Deviation between a measured or computed value (of an output variable) and the true, specified, or theoretically correct value.

- *Disturbance*: An unknown (and uncontrolled) input acting on a system.

- *Missed detection*: when there is not indication of a fault, though a fault has occurred.

- *Perturbation*: An input acting on a system, which results in a temporary departure from the current estate.

- *Residual*: Fault information carrying signals, based on the deviation between measurements and the model based computations.

- *Symptoms*: A change of an observable quantity from its normal behaviour.

About the functions

- *Fault detection*: Determination of faults present in a system at a particular time.

- *Fault isolation*: Determination of the type, location, and time of detection of a fault. Follows fault detection.

- *Fault diagnosis*: Determination of the kind, size, location and time of occurrence of a fault. Fault diagnosis includes fault detection, isolation and estimation.

- *Monitoring*: A continuous real-time task of determining the conditions of a physical system, by recording information, recognising and indicating anomalies in the behaviour.

- *Residual computation*: residual value is computed from the known variable.

- *Residual evaluation*: the residual is evaluated in order to detect, isolate and identify faults.

- *Supervision*: Monitoring of a physical system and taking appropriate actions to maintain the operation in the presence of faults.

About the models

- *Qualitative model*: A system model that describes the behaviour and relationships among system variables and parameters in heuristic terms such as causalities of if or then rules.

- *Quantitative model*: A system model that describes the behaviour and relationships among system variables and parameters in analytical terms such as differential equations.

- *Diagnostic model*: A set of static and dynamic relationships which link specific input variable -the symptoms- to specific output variables -the faults.

- *Analytical redundancy*: Use of two or more (but not necessary identical) ways to determine a variable where one way uses a mathematical process model in analytical form.

About the system properties and its measurements

- *Reliability*: Probability of a system to perform a required function under normal conditions and during a given period of time.

- *Safety*: Ability of a system not to cause any danger to people or equipment or environment.

2.2.1 Systems and controlled systems

A *system* is a set of interconnected components. Each of the components are chosen by the system engineer for achieving some function of interest. A function describes what the design engineer expects the components to perform. A given component performs its task because it has been designed for exploiting some physical principle(s), which in general are expressed by some relationship(s) between the evolution time of some system variables. Such relationships are called constraints, and the evolution time of the variables is called trajectory.

Some of the components may have been selected with the aim of controlling the process, i.e, being able to choose, between all the possible system trajectories, the one which will bring some expected result (achieve some given objective). Those components which allow to impose, or to influence, the trajectory of a given variable are called *actuators*. They establish some constraint between the variables of the process and some control variable, which is called control signal.

Actuators may be driven by human operators or by control algorithms. In both cases, closed loop control demands some information about the actual values of the system variables to be known. *Sensors* are components which are designed to provide this information. Thus a controlled systems is a set of interconnected components which include process components, actuators, sensors and control algorithms.

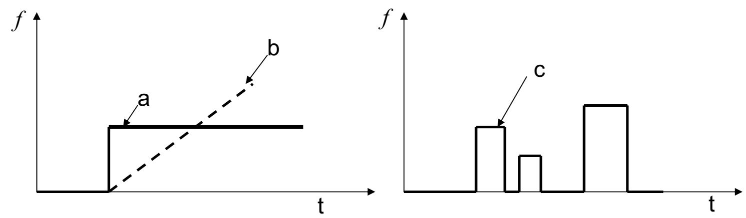


Figure 2.1: Time-dependency of faults: (a) abrupt; (b) incipient; (c) intermittent

2.2.2 Types of faults

The types of faults depend basically on the their location within the system, the number of components that can be affected and their temporal evolution.

Taking into account the effects of the faults, there are classified as *additive faults* (those which correspond to sensor and actuator faults) and *multiplicative faults* (or parametric):

- *Additive process faults*: These are unknown inputs acting on the plant, which are normally zero and which, when present, can cause a change in the plant outputs independent of the known inputs.

- *Multiplicative process faults*: These are changes (abrupt or gradual) in some plant parameters. They may cause changes in the plant outputs which also depend on the magnitude of the known inputs. Such faults describe the deterioration of the plant equipments, such as contamination, clogging, or the partial or total loss of the power.

The fault location can be distinguished in:

- *Sensor faults*: These are discrepancies between the measured and the actual values of the individual plant variables.

- *Actuator faults*: These are discrepancies between the input command of an actuator and its actual output.

- *Plant faults*:such faults change the dynamical properties of the system.

Regarding the time dependency of faults, they can be distinguished in Figure 2.1:

- *Abrupt faults*: These are faults that appear "abruptly" in a time instant. For example in a power supply break down.

- *Incipient faults*: These are faults that increase steadily and that are brought about by wear.

- *Intermittent faults*: These are faults that do not appear continuously.For example an intermittent electrical contact.

2.2.3 Approaches for fault detection and diagnosis

The methodology used in fault detection and diagnosis is clearly dependent on the process and the kind of available information. Existing approaches range from analytical methods to artificial intelligence and statistical approaches. From a modelling perspective, there are methods that require accurate process models, semi-quantitative models, or qualitative models. On the other hand, there are methods that do not assume any form of model information and rely only on historic process data. See (Venkatasubramanian et al., 2003) for a good review of approaches for FDD. In this work we have divided the methods within two categories: model-based techniques and non-model based techniques. They are described in the following sections.

2.3 Model-Based Techniques

The model-based diagnosis (MBD) approach rests on the use of a explicit model of the system to be diagnosed. The occurrence of a fault is captured by discrepancies between the observed behaviour and the behaviour that is predicted by the model. A definitive advantage of this approach is that it only requires knowledge about normal operation of the system, following a consistency-based reasoning method.

Two distinct and parallel research communities have been using the MBD approach. The fault detection and isolation (FDI) community has evolved in the automatic control field from the seventies and uses techniques from control theory and statistical decision theory. It has now reached a mature state and a number of very good surveys exist in this field (Control-Eengineering-Practice, 1997; Frank, 1996; Gertler, 1991; Iserman, 1997; Patton and Chen, 1991).

The diagnostic (DX) community emerged more recently, with foundations in the fields of computer science and artificial intelligence (De Kleer et al., 1992; De Kleer and Williams, 1987; Hamscher et al., 1992; Reiter, 1987). Although the foundations are supported by the same principles, each community has developed its own concepts, tools and techniques guided by their different modeling backgrounds. The modeling formalisms call indeed for very different technical fields; roughly speaking analytical models and linear algebra on the one hand and symbolic and qualitative models with logic on the other hand.

Most of FDI methods rely on the concept of *analytical redundancy* (Chow and Willsky, 1984; Gertler, 1991), next subsections describe this concept and its use for fault detection and isolation.

2.3.1 Analytical redundancy for fault detection and isolation

Consider the deterministic system modelled by

$$\dot{x}(t) = f(x(t), u(t), \varphi(t)) \tag{2.1}$$

$$y(t) = g(x(t), u(t), \varphi(t)) \tag{2.2}$$

where $x(t) \in R^n, u(t) \in R^r, y(t) \in R^m$ and $\varphi(t) \in R^q$ are respectively the state, input, output and fault vector, and f and g are given smooth vector fields.

The system normal operation on a time window $[\alpha, \beta[$ is described by :

$$\varphi(t) = 0, \forall t \in [\alpha, \beta[\tag{2.3}$$

while the occurrence of a fault at time γ is associated with :

$$\exists \lambda > \gamma : \varphi(t) \neq 0, \forall t \in [\gamma, \lambda[\tag{2.4}$$

Analytical redundancy relations (ARR)

Analytical redundancy is based on successive derivations of the output signal (2.2) which, together with the repeated use of (2.1) produce the system

$$\bar{y}(t) = G(x(t), \bar{u}(t), \bar{\varphi}(t)) \tag{2.5}$$

where $\bar{y}(t)$ (and also $\bar{u}(t)$ and $\bar{\varphi}(t)$) is the vector obtained by expanding $y(t)$ with its derivatives $\dot{y}(t), \ddot{y}(t), \ldots$ up to the order j:

$$\begin{array}{rcl} y(t) & = & g_0(x(t), u(t), \varphi(t)) \\ \dot{y}(t) & = & g_1(x(t), \bar{u}^{(1)}(t), \bar{\varphi}^{(1)}(t)) \\ \vdots & \vdots & \vdots \\ \bar{y}^{j}(t) & = & g_j(x(t), \bar{u}^{(j)}(t), \bar{\varphi}^{(j)}(t)) \end{array}$$

and G combines $g_0, g_1, \ldots, g_j$.

In a second step (2.5) is transformed into an equivalent system

$$\bar{y}(t) = G(x(t), \bar{u}(t), \bar{y}(t), \bar{\varphi}(t)) \iff \left\{ \begin{array}{c} G_1(x(t), \bar{u}(t), \bar{y}(t), \bar{\varphi}(t)) = 0 \\ G_2(\bar{u}(t), \bar{y}(t), \bar{\varphi}(t)) = 0 \end{array} \right. \tag{2.6}$$

where equations in subsystem G_2 are the so-called analytic redundancy relations (ARR), which are independent on the state. It can be shown that such ARR can always be found, provided the output can be derivated up to an order large enough (Chen and Patton, 1999; Gertler, 1998; Blanke et al., 2003). The interest of these ARR is obviously that - since the state has been eliminated - they depend only on the inputs, outputs, and faults, thus providing a means to check whether the no-fault hypothesis is consistent with the observed input - outputs.

Practical determination of analytical redundancy relations

From a practical point of view, obtaining the set of equations G_2 in (2.6) from the original set (2.5) makes use of a projection operator when system (2.1), (2.2) is linear (this is the parity space technique, see e.g. (Chen and Patton, 1999; Blanke et al., 2003) and for more general cases, it rests on elimination theory (see e.g. (Staroswiecki and Comtet-Varga, 2001) for the case where (2.1), (2.2) is polynomial).

It is also worth to notice that ARR are not uniquely defined. Indeed, any (linear or non-linear) combination of analytical redundancy relations is also an analytical redundancy relation and this property can be exploited to improve fault isolation by expanding *signature matrices* with these new ARRs. Signature matrix is introduced in next subsection devoted to Fault Isolation.

Computation and evaluation forms

Let the subsystem G_2 be decomposed as

$$G_2(\bar{u}(t), \bar{y}(t), \bar{\varphi}(t)) = G_c(\bar{u}(t), \bar{y}(t)) - G_e(\bar{u}(t), \bar{y}(t), \bar{\varphi}(t))$$

where for all input/output pairs $u(t), y(t)$ associated with system (2.1), (2.2)

$$G_e(\bar{u}(t), \bar{y}(t), 0) = 0 \tag{2.7}$$

Then condition $G_2 = 0$ can be written

$$r(t) \triangleq G_c(\bar{u}(t), \bar{y}(t)) \tag{2.8}$$
$$r(t) = G_e(\bar{u}(t), \bar{y}(t), \bar{\varphi}(t)) \tag{2.9}$$

where $r(t)$ is the residual vector, and (2.8), (2.9) are respectively its computation form and evaluation form. The first one describes how the residual value is obtained from the system inputs and outputs. The latter describes how the resulting value depends on faults.

According to (2.8) and (2.9), the fault detection and isolation procedure is decomposed into two steps. The first one is *residual computation* where the residual value is computed from the known variables, using the computation form (2.8). The second step is *residual evaluation,* that includes *fault detection* and *fault isolation.*

Fault detection

Given a time window $[\alpha, \beta[$, the fault detection problem is defined as follows: given the residual $r(t), t \in [\alpha, \beta[$ select the most likely hypothesis between H^0_{system} and H^1_{system} where

$$\begin{aligned} H^0_{system} &: \varphi(t) = 0, \forall t \in [\alpha, \beta[\\ H^1_{system} &: \exists [\gamma, \lambda[\subseteq [\alpha, \beta[: \varphi(t) \neq 0, \forall t \in [\gamma, \lambda[\end{aligned}$$

Using (2.7) and (2.9) the simplest implementation of a fault detection procedure is obtained by checking the residual value against zero at each time t (by a slight abuse of notation, the time intervals $[\alpha, \beta[$ and $[\gamma, \lambda[$ are no longer mentioned):

$$[H^0_{system} \implies r(t) = 0] \iff [r(t) \neq 0 \implies H^1_{system}] \tag{2.10}$$

For the sake of simplicity, only perfect deterministic models have been considered so far which results in (2.10) being indeed true. However, measurement noise, unknown

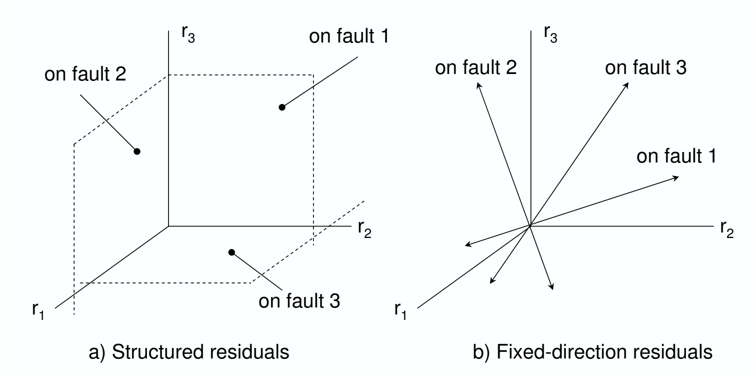


Figure 2.2: a) Structured residuals and b) Fixed-direction residuals

inputs, model uncertainties, will result in residuals being never exactly zero even in normal operation. This can be taken into account in a more realistic procedure which extends (2.10) as follows

$$[H^0_{system} \implies r(t) \in \mathcal{N}(0)] \iff [r(t) \notin \mathcal{N}(0) \implies H^1_{system}] \tag{2.11}$$

where $\mathcal{N}(0)$ is some neighborhood of zero. Note that false alarms - $r(t) \notin \mathcal{N}(0)$ under H^0_{system} - and missed detections - $r(t) \in \mathcal{N}(0)$ under H^1_{system} - are possible. The design of a set $\mathcal{N}(0)$ that guarantees both a low false alarm and a low missed detection rate is the central problem of statistical decision making (Basseville and Nikiforov, 1993; Brodsky and Darkhovsky, 2000; Nikiforov et al., 1996).

Fault isolation

Faults should not only be detected, but also be isolated, namely the faulty components should be determined. To facilitate de isolation, residual vector $r(t)$ is usually enhanced, in one of the following ways:

- *Structured residuals* (Figure 2.2a) . In response to a single fault, only a fault-specific subset of residuals becomes nonzero.
- *Fixed-direction residuals* (Figure 2.2b). In response to a single fault, the residual vector is confined to a fault specific straight line.

In FDI fault isolation is performed by means of an analysis of the *fault signature matrix*. The fault signature matrix Σ contains the dependency of a certain fault (column of the matrix) with each residual (row of the matrix). An element Σ_{ij} of this matrix is equal to 1 if the fault of the column j influences the residual of the row i, otherwise the element is equal to 0.

As was explained above, the fault detection procedure is obtained by checking each residual value $r_i(t)$ against zero at each time t. This procedure provides a set of *fault signatures* of the system, $s(t) = [s_1(t), \ldots, s_n(t)]$, where:

$$s_i(t) = \begin{cases} 0 & if \quad r_i(t) \in \mathcal{N}_i(0) \\ 1 & if \quad r_i(t) \notin \mathcal{N}_i(0) \end{cases} \tag{2.12}$$

Then, the fault isolation consists on finding which fault signature of the fault signature matrix is more similar to the signature $s(t)$ found experimentally. In order to measure a similarity between the two signatures, the Euclidean or the Hamming distances are usually used. For instance, if the Hamming distance is used, the procedure of fault isolation gives a distance vector for each fault signature $d(t) = [d_1(t), \ldots, d_n(t)]$, where:

$$d_j(t) = \sum_{i=1}^{n} (\Sigma_{ij} \oplus s_i(t)) \tag{2.13}$$

and $\oplus$ is the logical operator XOR. The theoretical fault signature that produces the smaller distance, indicates which fault is possibly affecting the system.

Example 1 (Fault signature)

Given a fault signature matrix:

Table 2.1: Fault signature matrix.

	f_1	f_2	f_3	f_4	f_5
ARR_1	1	0	1	0	0
ARR_2	0	1	0	1	0
ARR_3	0	1	1	0	1

if in a given time instant the observed fault signature is $s = [1, 0, 1]$, then the Hamming distance vector will be $d = [2, 1, 3, 0, 1]$. This allow to deduce, from choosing that fault that has the smallest theoretical signature distance to the observed fault, that the system is affected by the fault f_3.

For more details of structured residuals approach refer to (Chen and Patton, 1999; Staroswiecki and Comtet-Varga, 2001; Gertler, 1998; Hamelin et al., 1994). Structured residuals approach can be compared with the theory of logical diagnosis, as developed in the DX community. Detailed comparison and equivalence proofs have been given in (Cordier et al., 2004).

2.3.2 Fault detection and isolation techniques based on analytical models

The most used techniques for residual generations by means of analytical models are: observers (Chen and Patton, 1999), parity relations (Gertler, 1991), parameter estimation (Iserman, 1997) and structural analysis (Staroswiecki et al., 2000).

Observers

The basic idea of the observer or filter-based approaches is to estimate the states or outputs of the system from the measurements by using either Luenberger's observers in a deterministic setting (Beard, 1971; Frank, 1996) or Kalman's filters in a stochastic case (Willsky, 1976; Basseville, 1988). The flexibility in selecting observer gains has been studied (Frank and Ding, 1997). The freedom in the design of the observer can be used to enhance the residuals for isolation. The dynamics of the response can be controlled, within certain limits, by placing the poles of the observer.

Parity (consistency) relations

Parity equations are rearranged and usually transformed variants of the input-output or state-space models of the plant (Gertler, 1991; Gertler and Singer, 1990). The essence is to check the parity (consistency) of the plant models with sensor outputs (measurements) and known process inputs. The idea of this approach is to rearrange the model structure to get the best fault isolation. Parity relations concepts were introduced by (Chow and Willsky, 1984). Further developments have been made by (Gertler et al., 1990; Gertler et al., 1995; Staroswiecki and Comtet-Varga, 2001) among others.

There is a fundamental equivalence between parity relations and observer based methods. Both techniques produce identical residuals if the generators have been designed for the same specification (Frank, 1990; Gertler, 1991; Ding and Jeinsch, 1999).

Parameter estimation

The model-based FDI can also be achieved by means of using the system identification techniques if the basic structure of the model is known (Isermann, 1984; Iserman, 1997). This approach is based on the assumption that faults are reflected in the physical system parameters such as friction, mass, resistance, etc. The basic idea is that the parameters of the actual process are estimated on-line using well known parameter estimation methods and the results are compared with the parameters obtained initially under the fault-free case. Any discrepancy indicates a fault. The parameter estimation may be more reliable than the analytical redundancy methods, but it is also more demanding in terms of on-line computation and input excitation requirements.

A relationship has been found between parity relations and parameter estimation as well (Delmaire et al., 1994a; Delmaire et al., 1994b; Gertler, 1995; Gertler, 2000).

Structural analysis

The structural analysis is the study of the properties which are independent of the actual values of the parameters. Only links between the variables and parameters which result from the operating model are represented in this analysis. They are independent of the form under which this operating model is expressed (quantitative or qualitative data, analytical or non-analytical relations). The links are represented by graph on which a structural analysis is performed. The main advantages of the structural analysis approach are: it determines the part(s) of the system on which some ARR can be generated, and it is used to obtain the calculation sequences of the ARR.

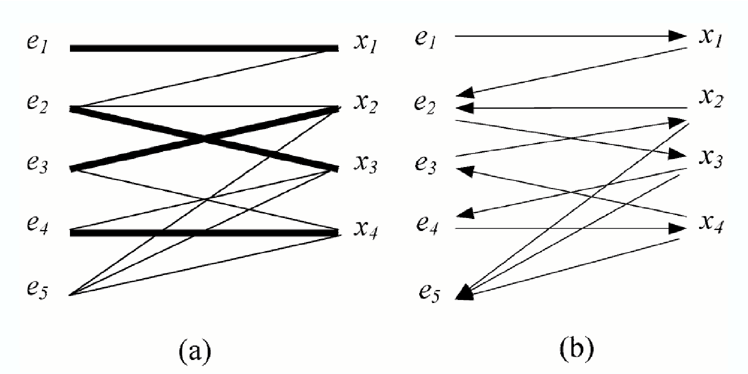


Figure 2.3: (a) Bipartite graph $G = (E \cup X, A')$ and complete matching of X into E indicated by bold arcs. (b) RPG indicating that e_5 is a redundant relation.

The structure of the system is obtained from a normal behavior model $S = (E, V)$, defined by a set of m relations E in which a set of n variables V are involved. The structure of a model can be represented by a structural matrix that crosses model relations in rows and model variables in columns, or equivalently by the bipartite graph $G = (E \cup X, A)$, where A is a set of arcs such that $a(i, j) \in A$ iff variable $v_j \in V$ appears in relation $e_i \in E$.

The set of variables V can be partitioned as $V = X \cup O$, where O is the set of observed (measured) variables, and X is the set of unknown variables. Then, structural approach of (Cassar and Staroswiecki, 1997) is based on determining a complete matching M between E and X in the bipartite graph G. A matching on a graph G is a set of edges of G such that no two of them share a vertex in common. A complete matching between E and X in a bipartite graph $G = (E \cup X \cup O, A)$, or equivalently in $G = (E \cup X, A')$, where A' is a subset of A, is one that saturates all of the vertices in E or X.

It corresponds to a selection of line-independent entries, i.e., not in the same row or column, in the structural submatrix crossing E and X. If the relation e_i is associated to the variable x_j by M, then e_i can be interpreted as a mechanism for solving for x_j. The resolution process graph (RPG) is defined as the oriented graph obtained from G by orienting the edges of A from x_j toward e_i if $a(i, j) \notin M$ and from e_i toward x_j if $a(i, j) \in M$. It provides the orientation of calculability (or causal interpretation) associated to M. The determination of M must account for the possibly restricted causal interpretation of some relations, e.g., a given relation may not be invertible and, hence, can only be used in a predefined direction. In practice, this is performed by orienting the corresponding edges a priori.

In (Cassar and Staroswiecki, 1997), it is shown that this graph can be used to derive the ARR. ARRs exist if and only if the number of relations $card(E)$ is strictly greater than the number of unknown variables $card(X)$. In this case, the complete matching is of X into E, and ARRs correspond to the relations that are not involved in the complete matching and, consequently, are not needed to determine the values of the unknown variables. These "extra-relations" appear as sink nodes of the RPG. ARRs of the form $r = 0$, where r is the residual of the ARR, are obtained from the extra relations by replacing the unknown variables with their formal expression in terms of observable variables, tracing back the analytical paths defined by the RPG. If $card(E) = card(X)$, then there are no ARRs, and the system is said to be nonmonitorable. The above is illustrated in Figure 2.3 with a toy example of five relations e_1 to e_5 and four unknown variables x_1 to x_4.

Appendixes A,B and C present the application of the structural analysis of two different systems that are going to be used in next chapters for FDI purpose. For more details of structural analysis approach refer to (Blanke et al., 2003).

2.3.3 Fault diagnosis based on qualitative-model

In this techniques the knowledge is obtained from the structure and the behavior of the process. Contrary to the analytical model, the qualitative models can be incomplete or contain uncertainties. In the last years there has been an important growth of contributions coming from the Artificial Intelligence community (this community uses the acronym DX for referring to model-based diagnosis), (Kuipers, 1994; Travé-Massuyès et al., 2001; Puig et al., 2002). The soft-computing community has also contributed in this field, where neural network (De la Fuente and Represa, 1997; Villegas and De la Fuente, 2006), fuzzy logic (Sauter et al., 1994; Mendoça et al., 2008), genetic algorithms (Calado and Sà da Costa, 1999) and multi-agent systems (Mendes et al., 2006; Sà da Costa et al., 2007) are applied.

2.4 Non Model-Based Techniques

In non model based techniques, the previous experimental records are analyzed in order to detect irregularities which would link the observed data (the symptoms) with the final conclusions (the diagnosis). We have divided the non model-based techniques within two categories: signal-based approaches and knowledge based approaches.

2.4.1 Signal-based approaches

Proper signals or symptoms are extracted from the system, which carry significant information about the fault of interest. The symptoms are used, directly or after proper modifications, for fault diagnosis. Typical symptoms are: the magnitudes of the time functions of measured signals, limit values, trends, statistical moments of amplitude distribution or envelope, spectral power densities or frequency spectral lines, correlations coefficients, covariance and so on.

There are numerous approaches of signal-based methods; some of them are as follows:

Physical redundancy

In this approach, multiple sensors are installed to measure the same physical variable. Any discrepancy among them indicates a sensor fault. Fault isolation is not possible with only two parallel sensors. A voting scheme can be formed with three sensors, which isolates the faulty sensor. Physical redundancy involves extra hardware cost and extra weight, the latter representing a serious concern in aerospace applications.

Special sensors and soft sensors

They may be installed explicitly for detection and diagnosis. These may be limit sensors (measuring e.g. temperature or pressure), which perform limit checking in hardware (see below). Other special sensors may measure some fault-indicating physical quantity, such as sound, vibration, elongation, etc.

Limit checking

In this approach, which is widely used in practice, plant measurements are compared to fixed thresholds; exceeding the limits indicates a fault situation. In many systems, there are two levels of limits, the first serves as pre-warning, while the second level triggers emergency actions.

Frequency analysis of plant measurements

Some plant measurements have a typical frequency spectrum under normal operating condition; any deviation from this is an indication of an abnormality. Certain types of faults may even have a characteristic signature in the spectrum that can be used for fault isolation.

Statistical methods

Statistical Process Control (SPC) is the use of statistical techniques to analyze a process or its outputs in order to take appropriate actions to ensure stable levels of quality within the process. The statistical methods are intended to provide an early detection. The calculation provides an early warning indicator of the effects on the process, allowing for correction at the best possible moment. For example, an instrument is used to take measurements on a certain quality variable. If the measurement is within the control limits, it is assumed that the instrument is working appropriately. If the measurement falls outside the control limits, the instrument is statically out of control. The majority of our processes are multivariate in nature, then a Multivariate SPC (MSPC) are to be used. Refer to (Basseville and Nikiforov, 1993) for more details on statistical methods.

2.4.2 Knowledge-based approaches

Fault diagnosis is a complex decision-making process which is a typical area of artificial intelligence. The knowledge type that are used to link observations with solutions can be classified into two forms: methods based on human experience and methods based on sets of cases with known solutions as their primary knowledge source.

Methods based on (direct encoding of) human experience

In many domains human experts are quite successful in finding cost-effectively solution to a diagnostic problem. However, such experts are usually rare and not always available. Since the beginnings of knowledge-based systems it has been a primary goal to capture their expertise in “expert systems”. The three main problems were (1) finding effective representations of the knowledge being close to the mental models of the experts, (2) organizing the process of transferring the knowledge from the expert into the system and (3) to process the encoded knowledge to infer the adequate solution.

During the last 30 years, a variety of knowledge representations and corresponding problem solving methods with many variants have been developed, each with different drawbacks as well as advantages and success-stories. The best process of knowledge transfer is still controversial. While some argue for a knowledge engineer as mediator between expert and system, others propagate knowledge acquisition tools enabling experts to self-enter their knowledge. The latter is usually much more cost effective, but requires tailoring the knowledge representations and the acquisition tools to the demands of the experts (Gappa et al. 1993). If experts switch between different methods, they should be supported in reusing as much knowledge as possible (Puppe 1998). In this section, we present a variety of representations for direct encoding of human expertise allowing for rapid building of diagnostic systems (for more details, see (Puppe et al. 1996)).

- *Decision trees*, where the internal nodes correspond to questions, the links to answer alternatives, and the leaves to solutions.

- *Decision tables* consisting of a set of categorical rules for inferring solutions from observations.

- *Heuristic classification* using knowledge of the kind "if <observations> then <solution> with <evidence>", the latter estimated by experts. Solutions are rated according to their accumulated evidence.

- *Set covering or abductive classification* using knowledge of the kind "if <solution> then <observation> with <frequency>", the latter estimated by experts. Solutions are rated according how well they cover (explain) the observed symptoms.

Methods based on knowledge

A diagnostic case consists of a set of observations and the correct solution, e.g. a set of attributes (A) and values (V) together with a solution (S), i.e. ((A1 V1) (A2 V2)... (An Vn) S) or in vector representation with a fixed order of attributes: (V1 V2 ... Vn S). A large set of cases with a standardized and detailed recording of observations represents a valuable source of knowledge, which can be exploited with different techniques:

- *Statistical/probabilistic classification* (Bayes' Theorem, Bayesian nets) using knowledge about the a-priori probability of solutions P(S) and the conditional probability P(O/S) of observation O if solution S is present, where the probabilities are calculated from a representative case base. (Pearl 1988; Russell and Norvig 1995, Part V).

- *Neural classification* using a case base for adapting the weights of a neural net capable of classifying new cases. Important net topologies with associated learning algorithms for diagnostics are perceptrons, backpropagation nets and Kohonen nets, (Kulikowski and Weiss, 1991).

- *Inductive reasoning* techniques try to compile the cases into representations also used by experts to represent their empirical knowledge (methods based on human experience). Therefore, complement those techniques of knowledge acquisition. The most popular learning methods are the *ID3-family* (Quinlan, 1997) and the *star-methods* (Mitchel, 1997).

- Diagnosis using a case base together with knowledge about the domain encoded in the definition of a vocabulary, a metric between cases to retrieve them and for reuse or update them to infer a new diagnosis. In *case-based reasoning* methods the objective is to reason based on experience. It requires a memory model for representing, indexing and organizing past cases. When confronted with a new case, indices are used to retrieve similar past cases from memory and to decide which case is the closest one to the new case. When old cases do not perfectly match to the system to be diagnosed, their solution must be adapted to infer an appropriate diagnosis. The new diagnosed case can be added to the memory providing a learning capability to the system.

2.5 Problems in Fault Detection and Diagnosis

2.5.1 Knowledge acquisition and representation

A perfect analytical model represents the deepest and most concise knowledge of the process. However, analytical models are hardly available, in which case knowledge-based models are a realistic alternative allowing one to exploit as much knowledge about the process as available. An additional difficulty in knowledge-based methods is to translate the numerical values (data coming from the process) to qualitative data (symbols) that can be used with these techniques. Different techniques for converting numbers in symbols have been developed (Dague, 1995; Colomer, 1998; Travé-Massuyès and Dague, 2003).

2.5.2 Measurement noise

Monitored plants are subjected to random noise. As unknown inputs, these noises affect the residuals and interfere with the detection and isolation of faults. In general, this situation requires a decision process which involves testing the residual against thresholds or uncertainty regions. In many practical situations, only limited information is available a priori concerning the noise and therefore the thresholds have to be chosen empirically.

If the statistical properties of the noise, together with the way it affects the plant output are known, the fault detection problem can be formulated in the framework of statistical decision making. Usually, it is reasonable to assume that the residuals are the summation of two components, one caused by the noise (which is random with mean zero) and other by the faults (which is deterministic but unknown). Hence, the residuals in presence of a fault may be considered as random variables which mean is determined by the fault. The fault detection problem is then posed as testing for a zero mean hypothesis while the isolation problem becomes a decision among a set of alternative hypotheses.

The design of residuals insensitive to noise and, at the same time, that guarantees a low false alarm and a low missed detection rate is a challenging problem which has motivated a huge number of works in the statistic field (Basseville and Nikiforov, 1993; Nikiforov et al., 1996; Brodsky and Darkhovsky, 2000; Gertler, 1998).

2.5.3 Modelling uncertainties

As it has been mentioned above, there are techniques for fault diagnosis that use a model of the system to be diagnosed. As well there are techniques that use a qualitative model (from the AI community) or analytical model (from the FDI community). As these models are an abstraction of the real system, it is common that they have errors or uncertainties.

In some cases, it is possible to have an accurate model of the system but it is too complex and a simplified one is more appropriate for a task to be undertaken (fault diagnosis) (Bonarini and Bontempi, 1994). This happens, for instance, when a non-linear system is linearised around an operating point and, therefore, a linear model is used to represent it. This happens, also, when a low order model is used to represent the behaviour of a higher order system in a range of frequencies.

However, in many cases there are uncertainties or imprecisions that make it difficult, if not impossible, to obtain accurate models. Some sources of this inaccuracy of the models are: physical phenomena that are difficult to identify or predict; the parameters of the system can change across time due to unknown, unpredictable or difficult phenomena; the knowledge of the system is not complete because the real system can not be observed or does not exit yet.

The effect of modeling uncertainties on the residuals was examined in (Gertler et al., 1990; Gertler, 1998); where modeling uncertainties are considered as multiplicative disturbances. When uncertainties can not be represented with quantitative models, i.e. models in which the parameters are real numbers, other kind of modeling is needed to consider these uncertainties. Some types of modeling that can represent the uncertainty of the systems are qualitative reasoning (Kuipers, 1993; Kuipers, 2001), multimodeling (Bonarini and Bontempi, 1994; Chittaro et al., 1993), interval models (also called semi-qualitative or semi-quantitative modeling) (Armengol, 1999).

2.5.4 Time misalignments

Due to the growing complexity and spatial distribution of automated systems, communication networks have become the backbone of most control architecture. As systems are required to be more scalable and flexible, they have additional sensors, actuators and controllers, often referred to as field (intelligent) devices (Lee et al., 2001; Staroswiecki, 2005). Networked Control Systems result from connecting these system components via a communication network such as CAN (Controller Area Network), PROFIBUS or Ethernet.

An increasing amount of research addresses the distributed control of inter-connected dynamical processes : stability and control (Dong et al., 2004; Montestruque and Antsaklis, 2003; Montestruque and Antsaklis, 2004), decision, coordination and scheduling (Tipsuwan and Chow, 2003; Walsh and Hong, 2001), diagnosis of discrete event systems (Neidiga and Lunze, 2005), and fault tolerance (Jalote, 1994; El-Farra et al., 2005; Patankar, 2004). However, only a few studies of the impact of the communication network on the diagnosis of continuous systems have been recently published (Zhang et al., 2004; Ding and Zhang, 2005; Fang et al., 2006).

In Model-based Fault Detection and Isolation (FDI), a set of residuals that should be ideally zero in the fault-free case and different from zero in the faulty case, are designed (Chen and Patton, 1999; Gertler, 1998; Blanke et al., 2003). However, in practice, residuals are different from zero, not only because of measurement noise, unknown inputs, and modelling uncertainties but also because of transmission delays. Since no network can communicate instantaneously, data which are used in the residual computation do not represent the state of the system at the time of the computation. Instead, they represent the state of the system at some (often unknown) time prior to the computation. Moreover, each variable being possibly transmitted under a different transmission delay, the whole set of data that are used in the residual computation may even not be consistent with the system state at any moment prior to the computation. Therefore, residuals that should theoretically be zero in the non faulty case might create false alarms as the result of transmission delays.

See for instance Figure 2.4 which represents a typical PROFIBUS network, composed of several controllers, input/output cards and a computer acting as an OPC (OLE for process control) server operating in an Ethernet network. An OPC-client computer performs the monitoring and FDI tasks.

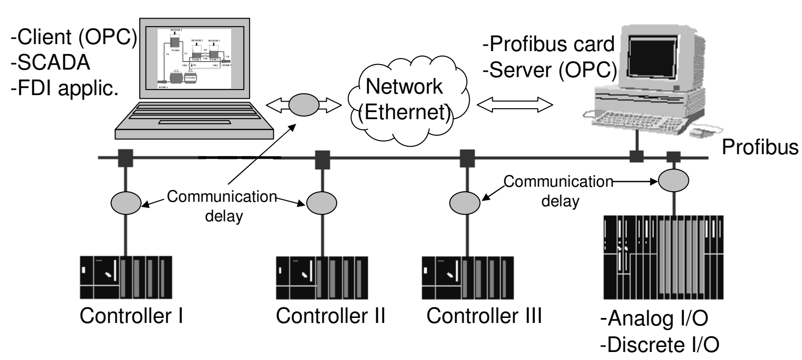


Figure 2.4: Data network delays in control systems

PROFIBUS interface modules transmit signal values, coming from the I/O card and controllers to the OPC server. The access to process data is performed by supervisory applications also under a client-server strategy with this OPC-server. Network and bus performances related to speed, availability of devices and parameter configuration of the field bus, location of sensors, conversion speed, sample or actualization rates and so on are factors than can influence the computation of residuals in a real application. One of the challenging problems that has motivated this work is the effect of transmission delays in the computation and evaluation of analytical redundancy based residuals.

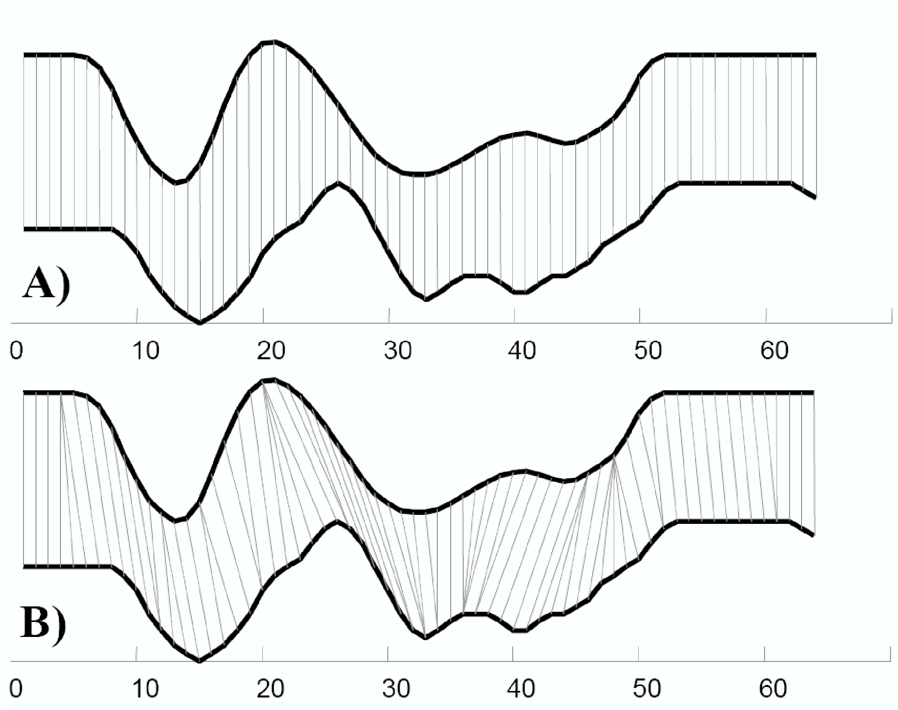


Figure 2.5: Two sequences that represent the measurements given by two sensors that are measuring the same process variable. A) erroneous comparison due to time misalignment; B) intuitive alignment feature

The comparison and matching of temporal signals (process measurements) for performing fault diagnosis can be also affected by the time misalignments within the measurements. For instance, in data-driven methods, as case base reasoning (knowledge based), a representative historical database of signals (cases) that have been previously analysed and suitably annotated, are used to identify the root cause of a change (fault) and develop an effective remedy (diagnosis). A difficult problem whit this method is to locate an instance (case) in the historical database (case base) that be the most similar to the specific data. Pattern classification or signal comparison is a popular method for finding similar signals in historical data. The challenge in this approach results from the fact that, because of the nature of industrial process, signals that result from two instances of the same change are not exact replicates. In others words, there are deviations between the two instances. The differences could be in the length (total time) of the two signals or in the magnitudes or profiles of the variables. Therefore, direct comparison of two signals would be incorrect, because there is no guarantee that the corresponding segments of the signals are being compared.

Figure 2.5A) depicts two sequences that represent the measurements given by two sensors that are measuring the same process variable. Notice that while the sequences have an overall similar shape, they are not aligned in the time axis. Therefore, the direct comparison (Euclidean distance) of the two sequences for diagnosis purpose can be affected by the misalignment of the signals. Euclidean distance, which assumes the i^{th} point on one sequence is aligned with i^{th} point on the other one, will produce a pessimistic dissimilarity measure. The vertical lines denote the corresponding i^{th} points that are being compared.

Figure 2.5B) shows the intuitive alignment when performing a nonlinear alignment for the signal comparison; oblique doted lines denote the corresponding i^{th} points that are being compared. Consequently, robust yet sensitive methods for comparing unsynchronized signals are an active area of research.

2.6 Conclusions

The basic aim of this chapter is to give some definitions and terminologies used in the field of fault detection and diagnosis and to review various approaches to fault diagnosis from different perspectives. Towards that goal, we have classified the different methods into two categories: (i) model-based methods (considering just analytical models); and (ii) non model-based methods, subdivided in: signal-based approach and knowledge-based approach.

We also present some problems that appear when performing fault diagnosis. Namely, knowledge acquisition and representation, noise in the measurements, model uncertainties and time misalignments.

The problem of time misalignments within the process variables has motivated this thesis, therefore the next chapter is devoted to explain its causes and effects in fault detection and diagnosis.

Chapter 3

Time Misalignments in Supervisory Systems

3.1 Introduction

The rapid development of communication technique promotes the widespread applications of networks, due to their flexibility, easy maintenance and low cost, building complex large-scale or teleautomatic systems. Instead to point-to-point connections, now sensors, actuators, as well as control and supervision stations exchange information trough networks which may also be used to other aims.

In Model-based Fault Detection and Isolation (FDI), a set of residuals that should be ideally zero in the fault-free case and different from zero, in the faulty case are designed. However, in practice, residuals are different from zero, not only because of measurement noise, unknown inputs, and modelling uncertainties but also because of transmission delays. Since no network can communicate instantaneously, data which are used in the residual computation do not represent the state of the system at the time of the computation. Instead, they represent the state of the system at some time prior to the computation. Moreover, each variable being possibly transmitted under a different transmission delay, the whole set of data that are used in the residual computation may even not be consistent with the system state at any moment prior to the computation. Therefore, residuals that should theoretically be zero in the non faulty case might create false alarms as the result of transmission delays.

Regarding the non model based techniques, that use the comparison and matching of signals for performing fault diagnosis, they can be affected by the unmatching of the signals due to a distortion in the time axis of one or both signals (time misalignments). For instance, in data-driven methods, a representative historical database of signals that have been previously analysed and suitably annotated, are used to identify the root cause of a fault . A difficult problem whit this method is to locate an instance in the historical database that be the most similar to the specific data. Pattern classification or signal comparison is a popular method for finding similar signals in historical data. The challenge in this approach results from the fact that, because of the nature of industrial process, signals that result from two instances of the same change are not exact replicates. In others words, there are deviations between the two instances. The differences could be in the length of the two signals or in the magnitudes or profiles of the variables. Therefore, direct comparison of two signals would be incorrect, because there is no guarantee that the corresponding segments of the signals are being compared.

This chapter is devoted to formulate the problem of time misalignment and communication delays from FDI and non model based fault diagnosis perspectives. It also gives the related work done in both direction. This chapter is organized as follows: Section 3.2 defines the concept of time misalignments and presents potential sources of time delays that produce time misalignments. Different models of network delays are given in Section 3.3. In Section 3.4 the related work of FDI in networked control systems (NCS) is presented. The effects of time misalignments in model based and non model fault detection and diagnosis are provided in Sections 3.5 and 3.6, respectively. Finally some concluding remarks are given in Section 3.7.

3.2 Time Misalignments

A time-series (or time sequence) is often defined as a series of values of a variable taken in successive periods of time. The instants in time at which the measurements are taken are known as time points. The length between time points can vary or be constant (sampling interval).

Time misalignment is the unmatching of two time-series due to a distortion (expansion or compression) in the time axis of one or both time-series. Figure 3.1 depicts two similar time-series whit the same mean and variance; note that while the sequences have an overall similar shape, they are not aligned in the time axis.

In process monitoring and fault diagnosis we deal with time-series (signals) generated by measurements given by the sensors located at the monitored system, this time-series are often affected by time misalignments. This time misalignments are caused by the **delays** within each measure (sample) given by the sensors, (Llanos et al., 2005).

3.2.1 Causes of time misalignments

Potential sources of time delays, or asynchronous availability of data, that produce time misalignments are:

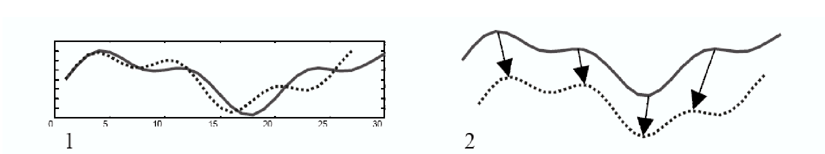


Figure 3.1: 1) Two similar time-series with the same mean and variance, 2) a possible feature alignment

Different time response among sensors

For example, pH-meters, temperature sensors or flowmeters have different time constants. This time delay is intrinsic to the measurement process.

Data communications

The delays on data communication system depend on the number of devices to be connected, the data transmission rate, throughput and reliability. Depending on the technology the time delays can be predictable or unpredictable. For instance, in a Profibus the major time delay is bounded from a parameter called (T_{TR}) -Target Rotation Time- (Vitturi, 2000). While for Ethernet the time delay is variable and cannot be predicted in advance due the interaction of several protocols, collisions and the traffic congestion.

Analog to digital conversion

Converters speed is not a real limitation. Nevertheless, some multichannel devices with a great number of multiplexed inputs could show significant limitations for applications operating at high sample rates.

Synchronous availability of data in remote applications

In a clients/server communication strategy there is not a certainty that the server can respond to synchronic petitions of clients with the same periodicity. In fact the server serves the last stamped data. This can cause the reception of repetitive data in a short period of time or misalignments in multiple-synchronous petitions.

Operative System (OS) latency

The minimum time needed for an operative system to serve interruption and to serve tasks under time constraints can be also a limitation. This consideration can be important when multiple devices (multiple OS) interoperating at the same time are coordinated to execute sequential tasks.

Sensor location

The configuration of the process itself may also give rise to time delay. Figure 3.2 shows one such example: the temperature of fluid in the tank is measured some distance along an outlet pipe rather than in the tank itself. If the temperature of the tank is changing a finite period of time will be required for fluid to be transported from the tank to the sensor. This time delay is intrinsic to the measurements process.

Table 3.1 summarizes typical time delays, (Trevelyan, 2004).

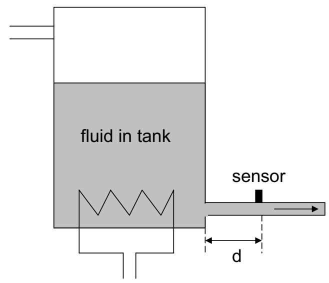


Figure 3.2: Tank with and temperature sensor. The placement of the sensor will result in an intrinsic time delay in measuring the tank fluid temperature, depending of the outflow rate

Table 3.1: Typical time delays, (Trevelyan, 2004).

A/D Conversion	1 μs (1MHz sampling rate)
	to 10 ms (1kHz sampling rate)
Memory access	0.1-10 μs (depending on
	bus traffic, memory access delay
O.S. latency	5-50 μs (real time O.S.)
	1-50 ms (Windows)
Commun. latency	1 ms (local LAN) to
(Ethernet)	1 s (between continents)
OPC server update	1 -2 s

3.3 Modelling of Network Delays

Delays have different characteristics depending on the network hardware and software. The network delay is typically varying due to varying network load, scheduling policies in the network and the nodes, and due to network failures. (Nilsson, 1998) proposed three models of the network delay:

- Constant delay,
- Random delay, which is independent from transfer to transfer
- Random delay with probability distributions governed by an underlying Markov chain (Grinstead and Snell, 1997).

3.3.1 Network modelled as constant delay

The simplest model of the network delay is to model it as being constant for all transfers in the communication network. This can be a good model even if the network has varying delays, for instance, if the time scale in the process is much larger than the delay introduced by the communication. In this case the mean value or maybe the worst case delay can be used in the analysis. If this is not the case, wrong conclusions can be drawn regarding system stability and performance.

One way to achieve constant delays is by introduction of timed buffers after each transfer. By making these buffers longer than the worst case delay time the transfer time can be viewed as being constant. This method was proposed in (Luck and Ray, 1990).

3.3.2 Network modelled as delays being independent

Network delays are usually random. The network delay can have several sources, for instance: waiting for the network to become idle; if several messages are pending, the wait can include transmission of the waiting messages; if transmission errors occur, a retransmission can be needed; in some networks collisions can occur if two nodes try to send at the same time, the resolution of this can include a random wait to avoid a collision at the next try.

As the activities in the system usually are not synchronized with each other, the number of the above listed delay causes that will occur is random. To take the randomness of the network delays into account in the model, the time delays can be modelled as being taken from a probabilistic distribution. To keep the model simple to analyze one can assume the transfer delay to be independent of previous delay times. In a real communication system the transfer time will, however, usually be correlated with the last transfer delay. For example, the network load, which is one of the factors affecting the delay, is typically varying with a slower time constant than the sampling period in a control system, i.e., the time between two transfers.

3.3.3 Network modelled using Markov chain

To model phenomena as network queues, and varying network loads, our network model needs to have a memory, or a state. One way to model dependence between samples is by letting the distribution of the network delays be governed by the state of an underlying Markov chain. Effects such as varying network load can be modelled by making the Markov chain do a transition every time a transfer is done in the communication network. For more details of this model refer to (Nilsson, 1998).

3.3.4 Network model adopted in this thesis

Since in this thesis we are focused in studying the effects of communication delays when fault detection and diagnosis is performed, the dependency between samples has not be considered; therefore network is modelled as delay being independent.

The normal operation of the communication network on a given time window $[\alpha, \beta[$ can be described by a very simple deterministic model. Namely the maximum delay Δ is assumed to be known:

$$
\begin{gathered}
[H^0_{network} \implies \forall t \in [\alpha, \beta[: \delta(t) \leq \Delta] \\
\Longleftrightarrow \\
\exists\, [\gamma, \lambda[\subseteq [\alpha, \beta[: \delta(t) > \Delta \implies H^1_{network}]
\end{gathered}
\tag{3.1}
$$

where $\delta_i \in R^+$ is the transmission delay which is time-dependent, generally is unknown and it is taken from a probabilistic distribution such a uniform or Poisson distributions (Johannessen, 2004).

3.4 Fault Diagnosis of Networked Control Systems

The purpose of this section is to present the related work of fault detection problem of networked control systems. Attention is focused on fault detection systems in case of time delays due to data transmission and in case of missing measurements due to data dropout. For more details of main ideas and result on fault diagnosis of NCS refer to (Fang et al., 2006; Fang et al., 2007).

3.4.1 Delays in networked control systems

Figure 3.3 depicts the block diagram of a networked control system and the delays that affect the system. There are essentially three types of delays: communication delay between the sensor and the controller, τ^{sc}; communication delay between the controller and the actuator, τ^{ca}; and computational delay in the controller, τ^{c}. τ^{c} is typically very short compared to τ^{sc} and τ^{ca}, therefore this delay is insignificant in many control techniques and could be included in τ^{ca}, (Nilsson, 1988). The sum of these three delays is referred to as the control network-induced delay.

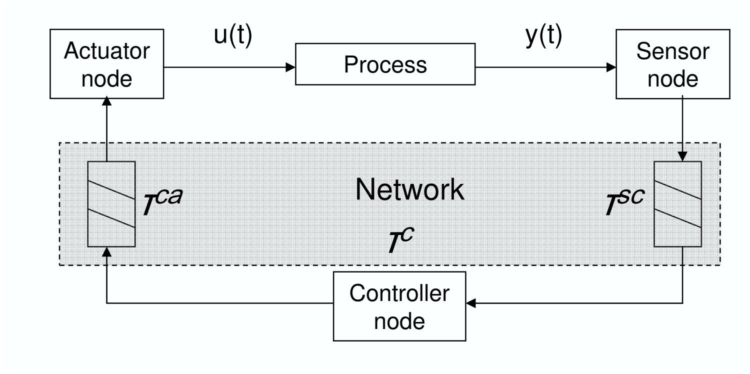


Figure 3.3: Delays in NCS: communication delay between the sensor and the controller, τ^{sc}; communication delay between the controller and the actuator, τ^{ca}; and computational delay in the controller, τ^{c} which could be included in τ^{ca}, (Nilsson, 1988).

A Networked Control Systems can be modelled by:

$$\dot{x}(t) = f(x(t), u(t-\tau^{ca}), \varphi(t)) \quad (3.2)$$
$$y(t) = g(x(t), u(t-\tau^{ca}), \varphi(t)) \quad (3.3)$$
$$\hat{y}(t) \triangleq y(t-\tau^{sc}) \quad (3.4)$$

where $\hat{y}(t)$ is the most updated output received by the controller.

It is evident that the networked-induced delay in NCS could influence the performance of traditional fault detection systems. For example, in (Ye et al., 2004; Ye et al., 2006) it is shown that due to the influence of the network-induced delay, an observer based fault detection system designed for a non-networked control system can't fulfill the basic requirement on residual generation anymore when it is used for a NCS, i.e., with the delay, the residual signal of a traditional residual generator will not be able to be decoupled from the control input any longer. (Sauter and Boukhbza, 2006) also studied the effect of unknown networked induced delays on conventional observer based residual generator. It is shown that the detection performances may be reduced due to the sensitivity of the residuals to the delays.

The effect of communication delay depend upon the communication topology. In distributed control systems the FDI unit can be located in different forms:

- located at the same node of the controller (Shanbin et al., 2006; Zhang and Ding, 2007). In this configuration the residual generator can directly obtain the control input $u(t)$ from the controller and it must get access to the measurement $y(t)$ through the network. Therefore residual will be affected just by the delay τ^{sc}.

- the FDI unit can be implemented as an algorithm in one node of the network (Llanos et al., 2007), see Figure 3.4. In this structure the residual computation can

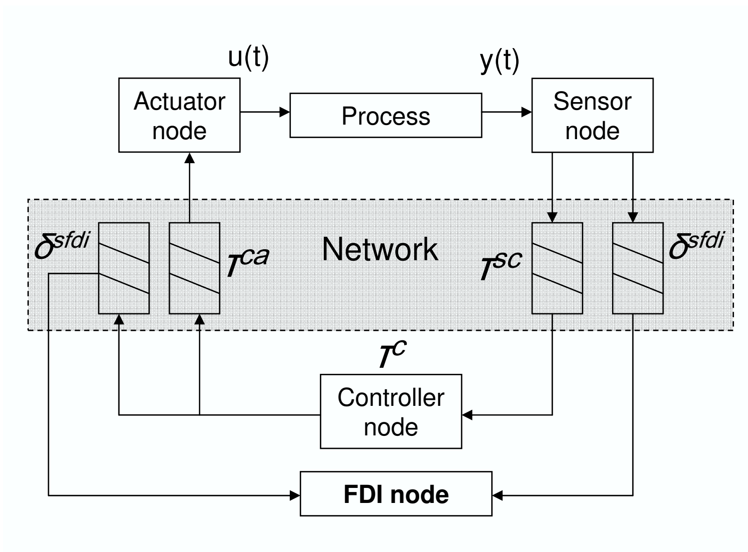


Figure 3.4: Structure adopted in this thesis: the FDI unit of the NCS is placed in a remote node.

be affected by the communication delay between the sensor and the FDI node, δ^{sfdi}; and the communication delay between the controller and the FDI node, δ^{cfdi}.

- and even, if we are dealing with large interconnected systems, the residual computation algorithm can be implemented following different architectures: centralized, decentralized or autonomous distributed, (Sauter et al., 2006).

This thesis is focused on the fault detection and diagnosis of NCS with not only network delays but also data dropout. **In this thesis the FDI unit has been implemented as an algorithm in one node of the network, see Figure 3.4.** Section 3.5 presents in more details the problem formulation of communication delays in residual computation under above mentioned implementation.

3.4.2 Related works in Fault Diagnosis of Networked Control Systems

In most of the related works the NCS were modelled as different simplified time-delay systems. Based on these models, many existed results, such as state observer and filtering theory, developed originally for time-delay systems could be utilized for fault diagnosis of NCS. **Despite we do not use time-delay systems models in this thesis, related works are here presented.**

Approaches based on state estimation and observer theory of time-delay systems

In this approaches the NCS model given by equation (3.2 and (3.4) is modified in order to take into account the delays in the state vector,$x(t)$:

$$\dot{x}(t) = f(x(t-\nu_i), u(t-\tau^{ca}), \varphi(t)) \quad (3.5)$$
$$y(t) = g(x(t-\nu_i), u(t-\tau^{ca}), \varphi(t)) \quad (3.6)$$
$$\hat{y}(t) \triangleq y(t-\tau^{sc}) \quad (3.7)$$

where ν_i denotes time delays in the state.

The work by (Yang and Saif, 1998) is the first paper to deal with the fault detection problem for time delay systems. A reduced-order unknown input observer was proposed to detect and identify the actuator and sensor faults for a class of state-delayed dynamic systems, in which the faults as well as other effects such as disturbances and higher-order nonlinearities were considered as unknown inputs. With some assumptions on the structure of system and distribution matrices, the completely unknown input decoupling from the residual is achieved.

In (Ding et al., 2002), a weighting transfer function matrix was first developed to describe the desired behaviour of residual respect to fault, and the observer-based fault detection filter for a class of linear systems with time-varying delays was designed in such a way that the error between the generated residual and fault (or, more generally weighted fault) is as small as possible in the sense of H_∞ -norm. The designing of the observer-based fault detection filter was then formulated into an H_∞ - model matching problem and, with the aid of an optimization tool, such as the linear matrix inequality technique, the problem has been solved.

The work in (Zhong et al., 2005) dealt with the fault detection problem for linear systems with L_2 -norm bounded unknown input and multiple constant time delays. Observer-based fault detection filter has been developed such that a robustness/sensitivity based performance index was minimized. The key idea of this study is the introduction of a new fault detection filter as residual generator and the extension of an optimization fault detection method for linear time invariant systems in (Ding et al., 2000) to the time-delay systems. A sufficient condition to the solvability of fault detection filter has been derived in terms of *Riccati* equation and a solution has been obtained by suitably choosing of a filter gain matrix and post-filter.

In (Zhong et al., 2004), the above mentioned optimization fault detection approach of (Ding et al., 2000) has been further modified to a class of neutral time-delay systems.

With a structure restriction on fault distribution, the work by (Jiang et al., 2002a; Jiang et al., 2002b; Jiang et al., 2003) developed an adaptive observer to the fault identification for both linear systems with multiple state time delays and a class of nonlinear systems

More recently, a new adaptive observer was proposed for the robust fault detection and identification of uncertain linear time-invariant systems with multiple constant time-delays in both states and outputs by (Jiang and Zhou, 2005).

In (Zheng et al., 2003b) they regard NCS as a sampled-data control systems with time-delays and set up its mathematical model. By constructing a memoryless reduced-order fault observer (Trinh and Aldeen, 1997), they achieve the fault detection of NCS. This observer-based fault detection approach for NCS is only serving as typical example.

With different assumptions and simplified systems models of NCS, variety existed works related to state estimation and observer for time-delay system can be extended to or used for NCS, and then for fault diagnosis algorithms for NCS were developed, such as the results presented in (Xie et al., 2005; Zheng et al., 2003a; Zheng et al., 2004; Zheng et al., 2005).

(Ding and Zhang, 2005) proposed an advanced observer-based scheme for the monitoring of distributed networked control systems. They do not attack directly the problem of delays but they consider that if the network load is minimised then the data transmission delays are reduced. Consequently, they design a two-level monitoring system. For each subsystem, a fault detection unit based on observer is embedded, which only makes use of local control input and measured output signals. In the global level, a central monitoring system is designed, which satisfies the required monitoring performance. Instead of collecting measured output signals from all subsystems, the inputs to the central monitoring system are the residual signals provided by the local fault detection units. In this way, a quantization can be used for transmission of signals. A reduced data transmission is thus achieved and the data transmission delays are expected to be smaller.

Approaches based on filtering

In this approaches the deterministic system model:

$$\dot{x}(t) = f(x(t), u(t), \varphi(t)) \tag{3.8}$$
$$y(t) = g(x(t), u(t), \varphi(t)) \tag{3.9}$$

is modified. (Ye et al., 2006) show that the unknown network-induced delay introduces an unknown additive item $h(t)$ into plant model:

$$\dot{x}(t) = f(x(t), u(t), h(t), \varphi(t)) \tag{3.10}$$
$$y(t) = g(x(t), u(t), \varphi(t)) \tag{3.11}$$

Therefore, when the network-induced delay is random, the unknown item $h(t)$ can also be regarded as a random disturbance. So a natural idea to reduce the influence of this disturbance is low-pass filtering the residual signal of a traditional generator. However, the idea could not be realized by only designing a traditional optimal residual generator first and then adding a low-pass filter to its output, because the optimization of the traditional residual generator does not mean that the new system consisting of it and a post-filter is still optimal. So they consider both parts of the new system (i.e. the residual generator and the low-pass filter) when designing the fault detection system.

A robust fault isolation filter for networked control systems with network-induced delays and multiple faults has been designed in (Shanbin et al., 2006). Where the communication delay between the sensor and the controller, τ_k^{sc} and the communication delay between the controller and the actuator, τ_k^{ca} are lumped together as a single delay, which is assumed to be random and governed by a Markov chain. Then, the effect of network-induced delays introduced into the control loop is regarded as time-varying disturbance. Based on this model, a fault isolation filter for fault detection of such networked control systems is then parameterized in order to generate the residuals having directional properties in response to a particular fault and decoupling from the disturbance.

Based on the study on the frequency domain characteristics of parity space approach (Ye et al., 2000; Zhang et al., 2006), a new fault detection approach based on parity space and Stationary Wavelet Transform (SWT) was proposed (Ye et al., 2004), adding SWT filters to an ordinary parity space based residual generator and considering both, the residual generators and the filters, in optimal design.

Fault detection of networked control systems with missing measurements

Another problem, close to data delays, also introduced by the networks is the data dropout. (Zhang et al., 2004) studied the fault detection problem of networked control systems with missing measurements due to data dropout. First, in order to cope with missing measurement, the structure of standard model-based residual generator is modified and dynamic network resource allocation is suggested. The dynamics of the residual generator is shown to be characterised by a discrete-time Markovian jump linear system. Then a residual evaluation scheme is developed aiming to reduce false alarm rate caused by missing measurement. Further, they proposed a co-design approach of time-variant residual generator and threshold to improve the dynamics and the sensitivity of the fault detection system to the faults. Related to this work is also the work on filtering in case of missing measurements. A detailed up-to-date summary of research in this direction could be found in (Wang et al., 2003).

3.5 Time Misalignments in Residual Computation

3.5.1 Problem formulation

Consider the deterministic system modelled by

$$\dot{x}(t) = f(x(t), u(t), \varphi(t)) \tag{3.12}$$

$$y(t) = g(x(t), u(t), \varphi(t)) \tag{3.13}$$

where $x(t) \in R^n, u(t) \in R^r, y(t) \in R^m$ and $\varphi(t) \in R^q$ are respectively the state, input, output and fault vector, and f and g are given smooth vector fields.

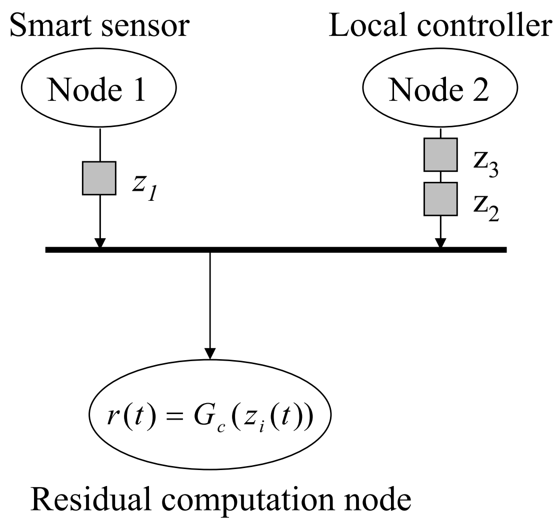


Figure 3.5: Data decomposition in distributed system

Data decomposition in distributed systems

In distributed control systems, the residual computation form is implemented as an algorithm in one node of the network. At each time t its input data are noted as

$$z(t) \triangleq \begin{pmatrix} \bar{u}(t) \\ \bar{y}(t) \end{pmatrix}$$

where $\bar{y}(t)$ (and also $\bar{u}(t)$) is the vector obtained by expanding $y(t)$ with its derivatives $\dot{y}(t), \ddot{y}(t), \ldots$ up to some derivation order.

According to the overall system distributed architecture, $z(t)$ is decomposed into a set of subvectors $z_i(t), i \in I$ and the computation form of the residual vector writes

$$r(t) = G_c\left(z_i(t), i \in I\right) \tag{3.14}$$

The subvectors $z_i(t)$ are such that all variables in $z_i(t)$ are transmitted in one single packet through the communication network. Note that this does not imply that all the variables produced at a given node are transmitted in one single packet. As shown in Figure 3.5

$$z(t) = (z_1^T(t), z_2^T(t), z_3^T(t))^T$$

where z_1 is produced and transmitted by node 1 (a smart sensor), while $z_2 \cup z_3$ are produced by node 2 (a local controller). The data are decomposed into packet z_2 and packet z_3 for their transmission through the communication system.

As result, the residual vector (3.14) designed under the assumption of perfect communication might create false alarms due to transmission delays.

The incidence of transmission delays

Due to the transmission delays the data $z_i(t)$ generated at the production nodes and the data $\hat{z}_i(t)$ available at the residual computation node must be distinguished. One obviously has

$$\hat{z}_i(t) = z_i(t - \delta_i), i \in I \tag{3.15}$$

where $\delta_i \in R^+$ is the transmission delay i.e. the data z_i was produced at time $t - \delta_i$, and it was received only at time t. Transmission delays δ_i may be time-dependent and they generally are unknown. The normal operation of the communication network on a given time window $[\alpha, \beta[$ can be described by a very simple deterministic model. Namely the maximum delay Δ is assumed to be known:

$$\begin{gathered}[H^0_{network} \implies \forall i \in I, \forall t \in [\alpha, \beta[: \delta_i \leq \Delta] \\ \iff \\ [\exists i \in I, \exists\, [\gamma, \lambda[\subseteq [\alpha, \beta[: \delta_i > \Delta \implies H^1_{network}]\end{gathered} \tag{3.16}$$

If communication delays are not taken into account residual computation can be performed as

$$r(t) = G_c(\hat{z}_i(t), i \in I) \tag{3.17}$$

but using data which are taken from the system at different time instants would obviously result in false alarms.

Taking into account communication delays by using future values of the arguments, as in (3.18), is obviously impossible.

$$r(t) = G_c(\hat{z}_i(t + \delta_i), i \in I) \tag{3.18}$$

A possible solution for decreasing the false alarms rate due to communication delays is by increasing the decision threshold, at the cost of reducing the sensitivity to faults.

Equation (3.18) has been used for considering the effect of communication delays in residual generation. If the analysis of the residual generator want also to be considered, equation (3.18) can be generated by any observer based residual generator which allows to control the dynamic of the residual.

In this thesis, two techniques aiming at the minimization of the false alarms caused by transmission delays without increasing the number of missed detection are proposed and explained in Chapters 4 and 6, respectively. The first one relies on the explicit modelling of communication delays and their most likely estimation, and the second one proposes an optimal dynamic alignment of data in a time window (Dynamic Time Warping).

3.6 Time Misalignments in Signal Based Fault Diagnosis

The time misalignments affect the non model based fault diagnosis particulary on the techniques that use the comparison and matching of temporal signals for diagnosis purpose.

3.6.1 Problem formulation

In non model based fault diagnosis the deterministic model given by equations (3.12) and (3.13) can not be considered anymore because function f and g are unknown. The only information we have from the process are the input $u(t)$ and the output $y(t)$ signals represented as time-series which are a rich source information that can be used for diagnosis purpose.

Pattern classification or signal comparison (Colomer, 1998; Webb, 2002) is a popular method for finding similar signals in historical data. The challenge in this approach results from the fact that, because of the nature of the industrial processes, signals that result from the two instances of the same change are not exact replicates. In others words, there are deviations between the the two instances. The differences could be in the length (total time) of the two signals or in the magnitudes or profiles of the variables. Therefore, direct comparison of two signals would be incorrect, because there is no guarantee that the corresponding segments of the signal are being compared.

Most of algorithm that operate with time-series of data use the Euclidean distance or some variation. However, Euclidean distance could produce an incorrect measure of similarity because it is very sensitive to distortions in the time axis (time misalignments), Figure (3.6) gives an example of time misalignments effects when comparing two time-series. Therefore, robust yet sensitive methods for comparing unsynchrinised signals are an active area of research.

3.6.2 Previous work in signal comparison for fault diagnosis

Due to significant advances in data collection and storage, vast amount of historical data is becoming commonly available. This data is a rich source of information about the process that can be used to improve plant operation. Multivariate statistics such as principal components analysis (PCA) have been widely used for process data classification and fault detection and diagnosis (Kano et al., 2001; Chen and Liao, 2002; Chiang and Braatz, 2003). PCA reduces the dimensionality of data with minimum loss of information by projecting high dimensional data onto uncorrelated vectors. The projections are chosen so that the maximum amount of information, measured in terms of its variability, is retained in the smallest number of dimensions.

(Krzanowski, 1979) defined the PCA similarity factor for measuring the similarity of

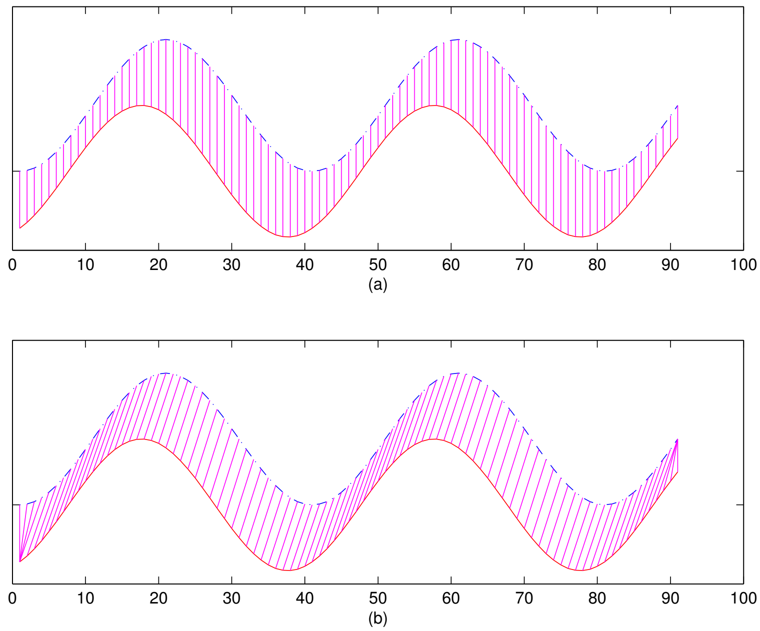


Figure 3.6: Two unsynchronised signals. a) erroneous comparison due to time misalignment; b) intuitive alignment feature

different groups of data. The PCA similarity factor compares the reduced subspaces:

$$S_{PCA} = \frac{1}{k}\sum_{p=1}^{k}\sum_{q=1}^{k}\cos^2\theta_{pq} \tag{3.19}$$

where θ_{pq} is the angle between the pth principal component of dataset S and the qth principal component of dataset T. (Raich and Cinar, 1997) used the PCA similarity factor for diagnosis process fault. (Singhal and Seborg, 2002) modified the PCA similarity factor by weighing the principal components with the square root of their corresponding eigenvalue, λ.

$$S_{PCA}^{\lambda} = \frac{\sum_{p=1}^{k}\sum_{q=1}^{k}\lambda_p^S\lambda_q^T\cos^2\theta_{pq}}{\sum_{p=1}^{k}\lambda_p^S\lambda_p^T} \tag{3.20}$$

A major limitation of the classical PCA-based approaches is that the PCA similarity factor is only applicable for stationary signals. To extend them to nonstationary signals, (Srinivasan et al., 2004) proposed a dynamic PCA-based similarity factor (S_{DPCA}^{λ}) that accounts for the temporal evolution of the signal. The main advantage of the PCA-based methods is their inherent ability to deal with multivariate signals and their low computational requirements. One strong assumption of PCA-based methods is that all batches have equal duration and all are synchronized.

Another family of data-driven approaches use signal comparison in order to overcome this. These approaches are based on the precept that the same types of faults show similar features in the process signal. This assumption is satisfied by most common normal and faulty operating states in processes. By comparing the online signal with a database of signals corresponding to the different fault classes, any fault in the process can be diagnosed. The challenge in these methods is that it is normal for two similar signals to be slightly different or misaligned and not match each other perfectly. This synchronization problem can be overcome using Dynamic Time Warping (DTW). In this thesis we proposed a slight modification of DTW in order to adapt it for on-line application. Chapters 6 and 7 are devoted to explain the main particularities of the new algorithm, as well as its application on a knowledge-based method for fault diagnosis, known as Case based Reasoning.

3.7 Conclusions

Control over data networks has many advantages compared with traditional control systems, such lower cost, greatly reduced wiring, weight and power, simpler installation and maintenance and higher reliability. However, the data dropout, limited bandwidth, time delay due to data transmission, asynchronous clock among network nodes and other peculiarities of networks could degrade the performances of the closed-loop systems and even destabilize them. Above problems have been intensively studied by the control community in the last several years including: analysis of impact of network on control performance, design of control algorithm taking into account the above factors and proposal of new network protocol suitable for control. However only a few studies of the impact of the communication network on the diagnosis of continuous systems have been published. Namely, in model-based Fault Detection and Isolation (FDI) a set of residuals is computed, that should be ideally zero in the fault-free case and different from zero, in the faulty case. However, in practice, residuals are different from zero, not only because of measurement noise, unknown inputs, and modelling uncertainties but also because of transmission delays introduced by the communication networks.

Most of the related works attack the fault detection problem of NCS with random time delay in two ways: extending or directly using the existed state estimation and observer theory of time-delay system to develop dedicated fault detection method for NCS, and developing fault diagnosis methods which are robust to an unknown item produced by the network-induced random time-delay. In both cases a modification of the system model in order to take into account the communication delays is introduced. This thesis proposes a solution for tame-delays effect in NCS when performing FDI. The solution doesn't modify the system model and is based on the explicit modelling of communication delays and on their best-case estimation. The technique is explained in next Chapter.

Concerning the FDI with data dropout, this thesis proposes a technique for false alarms reduction due to data dropout, the solution is based on the residual behaviour and the communication status. The technique is explained in Chapter 5.

Regarding the signal-based fault diagnosis, there are several techniques for comparing time-series for fault diagnosis purpose, however most of them do not explicitly consider the problem of time-misalignments. A method known as dynamic time warping (DTW) allows a nonlinear alignment within two signals. A shortcoming of this method is its expensive computational cost (in both time and memory) which make it normally useful only for off-line applications. This has motivated a slight modification of DTW in order to adapt it for on-line application. Chapters 6 and 7 are devoted to explain details about DTW, the main particularities of the new algorithm, as well as its application on Case based Reasoning.

Chapter 4

Delay Estimation for Residual Computation

4.1 Introduction

As has been seen en previous chapter residuals that should theoretically be zero in the non faulty case might create false alarms as the result of transmission delays. The false alarms rate can be decreased by increasing the decision threshold, at the cost of reducing the sensitivity to faults. In this chapter, a technique aiming at the minimization of the false alarms caused by transmission delays without increasing the number of missed detection is proposed. It relies on the explicit modelling of communication delays, and their most likely estimation (Llanos et al., 2006; Llanos et al., 2007).

The chapter is organized as follows: Section 4.2 presents the influence of transmission delays. The decision procedure under unknown transmission delays is analysed in Section 4.3. An optimization technique for the estimation of unknown delays is described in Section 4.4. Illustrative examples are shown in Section 4.4.1 and 4.5. Finally some concluding remarks are given in Section 4.6.

4.2 Influence of communications delays in residual computation

As was stated in section 3.5.1, in distributed control systems, the residual computation form is implemented as an algorithm in one node of the network.

If communication delays are not taken into account residual computation can be performed as

$$r(t) = G_c(\hat{z}_i(t), i \in I) \tag{4.1}$$

where $\hat{z}_i(t)$ is the data available at the residual computation node. Using data which are taken from the system at different time instants would obviously result in false alarms.

Taking into account communication delays by using future values of the arguments, as in (4.2), is obviously impossible.

$$r(t) = G_c(\hat{z}_i(t+\delta_i), i \in I) \tag{4.2}$$

Finally, the only possibility to obtain a feasible algorithm is to 'synchronize' the data by using a delay τ as follows :

$$\rho(t) = G_c(\hat{z}_i(t-\tau+\delta_i), i \in I) \tag{4.3}$$

where $\rho(t)$ is the residual available at the residual computation node. Note that at a time t one in fact computes the value that the residual had at time $t-\tau$

$$\rho(t) = r(t-\tau) \tag{4.4}$$

The delay τ must obviously satisfy

$$\tau \geq \max_{i \in I} \delta_i \tag{4.5}$$

4.3 The decision procedure under unknown transmission delays

When the vector of transmission delays

$$\delta = (\delta_i, i \in I)$$

is perfectly known, the decision procedure:

$$[H^0_{system} \implies r(t) \in \mathcal{N}(0)] \iff [r(t) \notin \mathcal{N}(0) \implies H^1_{system}] \tag{4.6}$$

where $\mathcal{N}(0)$ is some neighborhood of zero, can be directly run by choosing

$$\tau = \|\delta\|_\infty$$

from (4.4) it follows that

$$[H^0_{system} \implies \rho(t) \in \mathcal{N}(0)] \iff [\rho(t) \notin \mathcal{N}(0) \implies H^1_{system}] \tag{4.7}$$

however when $\rho(t) \notin \mathcal{N}(0)$ the fault detection process is delayed by τ.

When transmission delays are unknown, the decision has to be taken in the presence of the so-called *nuisance parameters* δ. From (3.16) and (4.7) the following decision logic is true

$$\begin{aligned}[H^0_{system} \wedge H^0_{network}] \implies \exists\delta : [(\|\delta\|_\infty \leq \Delta) \wedge (\rho(t) \in \mathcal{N}(0))] \iff \\ \forall\delta : [(\|\delta\|_\infty > \Delta) \vee (\rho(t) \notin \mathcal{N}(0))] \implies [H^1_{system} \vee H^1_{network}]\end{aligned} \tag{4.8}$$

This decision logic expresses that the non existence of a vector of transmission delays, δ such that (1) $\|\delta\|_\infty \leq \Delta$ and (2) the residual lies inside $\mathcal{N}(0)$ evidences that the system, the network, or both do not operate properly. Even though both faults in the network and in the system can be detected, they cannot be isolated from each other in the absence of extra information.

4.4 Estimating the transmission delays

Let the system fault detection neighborhood $\mathcal{N}(0)$ be defined by

$$\mathcal{N}(0) = \{\rho : J(\rho) \leq \sigma\}$$

where $\forall\rho \neq 0,\ J(\rho) > 0,\ J(0) = 0$, for example $J(\rho) = \rho^T Q \rho$ with $Q > 0$ and $\sigma > 0$ is a given decision threshold. Checking the property $H^0_{system} \wedge H^0_{network}$ can be done by solving the following optimization problem

$$\breve{\delta} = \arg \min_{\|\delta\|_\infty \leq \Delta} J(\rho(t)) \tag{4.9}$$

where $\rho(t) = G_c(\hat{z}(t-\tau+\delta))$. Let $\breve{\tau} = \left\|\breve{\delta}\right\|_\infty$, $\breve{\rho}(t) = G_c(\hat{z}(t-\breve{\tau}+\breve{\delta}))$ and $\breve{J}(t) = J(\breve{\rho}(t))$, then the following interpretation holds :

(1) $\breve{\delta}$ is the "most likely" vector of admissible delays in the sense that the function $\breve{J}(t)$ associated with the delayed residual $\breve{\rho}(t)$ is minimum

(2) This residual may or may not be compatible with the hypothesis that the system operates in a nominal way, and therefore the decision logic (4.8) becomes:

$$\breve{J}(t) > \sigma \implies H^1_{system}$$

Finally it should be noted that the estimation of $\breve{\delta}$ by (4.9) implements a sufficient condition-based decision logic. Indeed, if the minimal value $\breve{J}(t)$ associated with $\breve{\delta}$ does not satisfy $\breve{J}(t) \leq \sigma$ then no other estimation will do. However, the set $\{\delta : [\|\delta(t)\|_\infty \leq \Delta] \wedge [\rho(t) \in \mathcal{N}(0)]\}$ may contain more than one element.

4.4.1 Searching for a minimum

The cost function $J(\rho(t)) = J\left[G_c(\hat{z}(t-\tau+\delta))\right]$ is in general a nonlinear function of the adjustable parameters δ, and its minimum can be found using well known iterative search methods (Reklaitis et al., 1983). However, since the problem is to be solved in real time, it is of interest to study the conditions under which the estimation $\breve{\delta}$ can be found quickly and accurately. Note that even when a dynamic feedback control loop is involved, the on-line search for a minimum, being a part of the Fault Detection and Isolation algorithm, can be run at a much lower frequency than the one associated with the control computation, for systems where faults are not critical (should faults be critical, it can be assumed that the network would have been designed so as to make transmission delays negligible).

Persistent excitation condition

Assuming that all functions involved are differentiable, the solution of the optimization problem (4.9) satisfies the necessary condition :

$$\frac{\partial J(\rho(t))}{\partial \delta_i} + \mu_i = 0, \quad i \in I \tag{4.10}$$

where μ_i is the Khün and Tücker parameter associated with the inequality constraint $\delta_i \leq \tau$. This system can be solved for $\breve{\delta}$ if its Jacobian is not too ill conditioned in a neighborhood of the optimum. From

$$\frac{\partial J(\rho(t))}{\partial \delta_i} = \left[\frac{d\hat{z}_i}{d\delta_i}(t-\tau+\delta_i)\right]^T \left[\frac{d\rho(t)}{d\hat{z}_i}\right] \frac{dJ(\rho(t))}{d\rho}$$

it is seen that there are some system trajectories that produce a rank defective Jacobian, namely when $z_i(t)$ is constant for some $i \in I$. Therefore for delay estimation, it is necessary that a *persistent excitation* condition be satisfied, such that no transmitted packet of variables is constant over time.

When the persistent excitation condition is not satisfied, the previous estimated value of the delay can be used to compute $\rho(t)$.

Illustrative example 1

Let

$$r(t) = y_1(t) - \frac{1}{2}y_2(t) - \frac{1}{2}y_3(t) \tag{4.11}$$

be the residual computation form associated with three sensors $y_1(t)$, $y_2(t)$ and $y_3(t)$ that measure the same variable $x(t)$, but are located in three different nodes. The data available at the residual computation node are $\hat{y}_i(t), i = 1, 2, 3$, and the actually computed residual is

$$\rho(t) = \hat{y}_1(t-\tau+\delta_1) - \frac{1}{2}\hat{y}_2(t-\tau+\delta_2) - \frac{1}{2}\hat{y}_3(t-\tau+\delta_3)$$

where δ_i are the delays through the network and τ is their maximum value (4.5).

Simulations have been performed with $x(t) = sin(wt)$ as the unknown variable evolution, and with the following transmission model : under $H^0_{network}$, the transmission delays are uniformly distributed with a maximal value of $\Delta = 5s$.

Under $H^0_{system} \wedge H^0_{network}$ one obviously obtains $\rho\,(t) = 0$ when δ_i are replaced by their actual values. These values being unknown, the vector of admissible delays $\breve{\delta}$ is estimated, by solving the optimization problem

$$\min \rho^2(t), \text{under the constraint} \qquad \delta_i \in [0, 5]\,\mathrm{s} \tag{4.12}$$

Matlab has been used for simulation and optimisation. The function *fmincon* has been used for delays estimation. This function uses a sequential quadratic programming (SQP) method (Reklaitis et al., 1983), and it is suitable for finding a minimum of a constrained nonlinear multivariable function. Initial condition, $\delta^{(0)}$, for the algorithm has been fixed to zero.

In order to study the benefits of delay estimation to optimize residual computation, neither unknown inputs (noise or disturbances) nor uncertainties have been considered. The analysis has been focused on the performance of the method to reduce false alarms in the absence of faults. Figure 4.1 depicts the residual computed without and with the delay estimation in a fault free situation ($r(t)$ above and $\rho(t)$ below). The value of w was $0.05 rad.s^{-1}$. It is well known that false alarms can always be avoided (e.g. by never firing any alarm). This can be seen on the figure, which shows that false alarms are avoided in both cases, but at the cost of one order of magnitude on the decision threshold (± 0.19 instead of ± 0.02), which means worse performances when faults will be present (larger missed detection rate, bigger size for faults to be detectable).

It is worth noting that, at particular (periodical) times the residual $\rho(t)$ has significant deviations with respect to zero. Those times correspond with the maximal and minimal values of the sinusoidal signal $x(t)$. Indeed, in those cases the small variability of the signals does no guarantee the fulfillment of the persistent excitation condition. Nevertheless, even in that situation the residual $\rho(t)$ is one order of magnitude smaller than $r(t)$.

The error distributions between actual and estimated delays are depicted in the three histograms of Figure 4.1. The errors are close to zero, which means a good estimation. For the first delay δ_1, there are less negative errors than for the other delays, this is due to the fact that δ_1 is associated with the largest term in the residual. Consequently, the use of gradient methods in the optimization procedure forces a faster adjustment. It is important to remark that the uncertainty reflected in the comparison does not affect the computation of the residual $\rho(t)$ and it is just relevant when the persistent excitation condition is not fulfilled.

Local minima

The search for a minimum is started out from an initial guess on the parameters $\delta^{(0)}$ and in general it will converge towards a local minimum. Starting with zero at the very beginning and taking the last estimate of the transmissions delays as the initial guess for the next estimation seems to be a good approach when the exploitation conditions of the network do not change at a faster rate than the rate at which residuals are computed. Converging towards a local minimum is a source of false alarms, since the estimation $\breve{\delta}$ may lead to $\breve{\rho}(t) \notin \mathcal{N}(0)$ while the global minimum, say δ^*, would have provided $\rho^*(t) \in \mathcal{N}(0)$. In special cases, such as linear systems and convex cost functions, global minima will be found, but in more general cases, algorithms which avoid getting trapped in a local minimum are to be used (Goldberg, 1989).

False alarm vs missed detection

Given certain threshold values, statistical hypothesis theory can be applied to predict the false alarm and missed detection rates based on the statistics of the data. Increasing the threshold (shifting the vertical line to the right in Figure 4.2a) decreases the false alarm rate but increases de missed detection rate. Attempts to lower the false alarm rate are usually accompanied with an increase in the missed detection rate, with the only ways

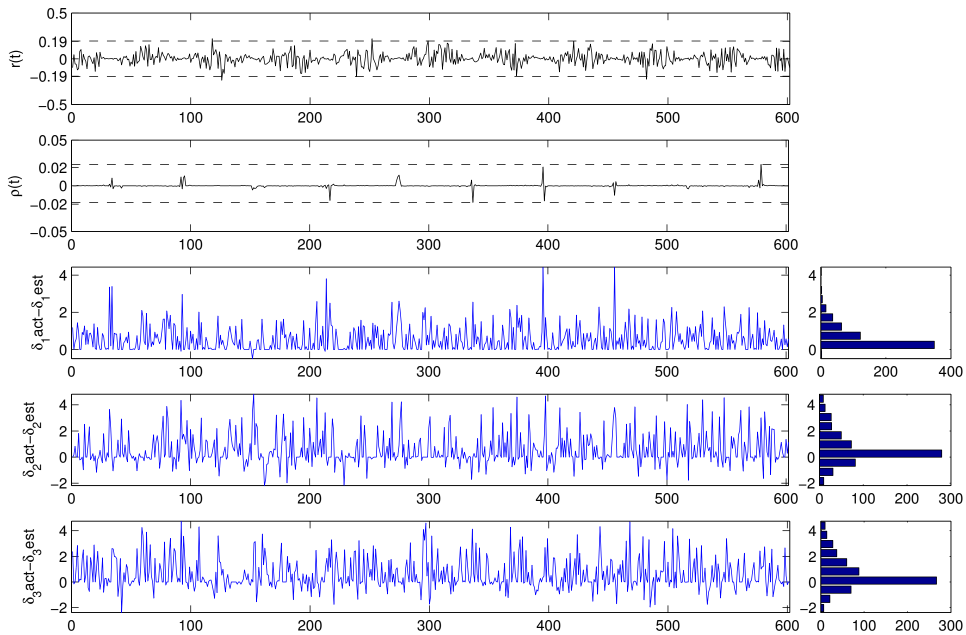


Figure 4.1: Residual in fault free situation and comparison of actual and estimated delays

to get around this tradeoff being to collect more data, or to reduce the normal process variability (e.g., through installation of sensors of higher precision). The value of the type I error (false alarms), also called the level of significance λ, specifies the degree of tradeoff between the false alarm rate (Type I error) and the missed detection rate (Type II error) when deciding the presence of faults in the system.

The proposed optimisation technique aims to minimise the false alarm rate caused by transmission delays without increasing the missed detection rate. Figure 4.2b) shows the intuitive idea.

Illustrative example 2

Continuing with example 1, table 4.1 gathers the decision thresholds that must be selected in order to avoid false alarms, for increasing values of w. It is seen that the larger w, the larger the threshold. Fixing the bandwidth of the communication channel and increasing the frequency of the signal to be transmitted decreases the quality of the transmission. It can be noted that by choosing larger thresholds to avoid false alarms, one can expect the missed detection rate during faulty situations to be deteriorated, as can be seen below.

Table 4.2 and table 4.3 present the missed detection rate associated with different additive faults (bias values of $0.05, 0.1, 0.15$ and 0.2) on residuals $r(t)$ and $\rho(t)$ of example 1. And different signal dynamics (w values of $0.05, 0.11, 0.15, 0.21$ and $0.31 rad.s^{-1}$). Note that in spite of the improvement associated with the estimation of the transmission delays,

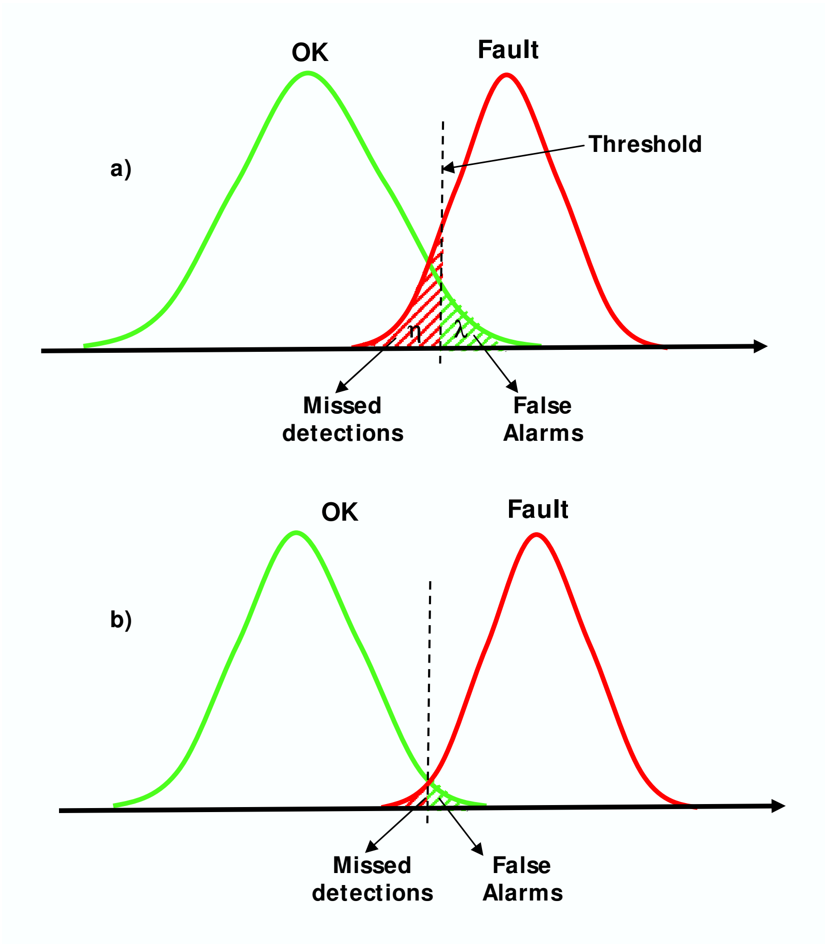


Figure 4.2: a)Level of significance, missed detections (Type II error) and false alarms (Type I error). b)False minimisation without increasing the missed detections.

missed detections are still possible when using residual $\rho(t)$, but they are much less than the missed detections associated with residual $r(t)$. In other words, the minimum size of detectable faults can be reduced using the residual $\rho(t)$ instead of residual $r(t)$ without increasing the false alarms rate.

4.5 Application to a control position of a DC motor.

Figure A.1 depicts the schematic diagram of a control position for a DC motor. The aim of this system is to control the position $c(t)$ of the mechanical load according to the reference position $u(t)$. This system has been taken from (Ogata, 1997).

The transfer function in open loop is:

$$\frac{C(s)}{E_v(s)} = \frac{K_m}{s(T_m s + 1)} \tag{4.13}$$

where, $C(s) = \mathcal{L}[c(t)]$ and $E_v(s) = \mathcal{L}[e_v(t)]$, $e_v(t)$ is the error between the output and the input positions. K_m and T_m are known parameters of the system.

Let

$$r(t) = T_m \frac{d^2c(t)}{dt^2} + \frac{dc(t)}{dt} - K_m e_v(t) \tag{4.14}$$

be the residual computation form associated with the system, where $c(t)$ and $e_v(t)$ are assumed to be produced in two different nodes of a distributed system, see Figure 4.4. The design of this residual is explained in more details in Appendix A.

The data available at the residual computation node are $\hat{e}_v(t)$ and $\hat{c}(t)$, and the actually computed residual is

$$\rho(t) = T_m \frac{d^2c(t-\tau+\delta_1)}{dt^2} + \frac{dc(t-\tau+\delta_1)}{dt} - K_m e_v(t-\tau+\delta_2) \tag{4.15}$$

where δ_i are the delays through the network and τ is their maximum value (4.5).

Simulations have been performed with $K_m = 5.5\text{s}^{-1}$ and $T_m = 0.13$s. $u(t)$ has been modelled as an unit step signal and $c(t)$ as its response.

In order to study the benefits of delay estimation to optimize residual computation, neither unknown inputs (noise or disturbances) nor uncertainties have been considered.

Table 4.1: Decision thresholds for $r(t)$ and $\rho(t)$.

	w_1 $0.05rad/s$	w_2 $0.11rad/s$	w_3 $0.15rad/s$	w_4 $0.21rad/s$	w_5 $0.26rad/s$	w_6 $0.31rad/s$
$r(t)$	± 0.02	± 0.04	± 0.06	± 0.08	± 0.1	± 0.13
$\rho(t)$	$\pm 5 \times 10^{-4}$	± 0.001	± 0.002	± 0.003	± 0.004	± 0.005

Table 4.2: Missed detection rate for $r(t)$.

bias value	w_1 $0.05rad/s$	w_2 $0.11rad/s$	w_3 $0.15rad/s$	w_4 $0.21rad/s$	w_5 $0.26rad/s$	w_6 $0.31rad/s$
0.05	17%	98%	93%	100%	100%	100%
0.1	0%	26%	66%	96%	96%	100%
0.15	0%	3%	17%	68%	80%	91
0.2	0%	0%	2%	23%	56%	86%

Table 4.3: Missed detection rate for $\rho(t)$.

bias value	w_1 $0.05rad/s$	w_2 $0.11rad/s$	w_3 $0.15rad/s$	w_4 $0.21rad/s$	w_5 $0.26rad/s$	w_6 $0.31rad/s$
0.05	0%	0%	0%	3%	0%	10%
0.1	0%	0%	0%	0%	4%	0%
0.15	0%	0%	0%	0%	0%	0%
0.2	0%	0%	0%	0%	0%	0%

The analysis has been focused on the performance of the method to reduce false alarms in the absence of system faults and to reduce the missed detection rates in presence of a sensor fault.

4.5.1 Considering delays in one signal

Let's consider that delays are introduced only for the signal $c(t)$, with the transmission model $H^0_{network} \iff \delta_1$, where δ_1 is uniformly distributed in the interval $[0, 0.1]\,\mathrm{s}$.

Under $H^0_{system} \wedge H^0_{network}$ one obviously obtains $\rho\,(t) = 0$ when δ_1 is replaced by its actual value. When this value is unknown, the admissible delay $\breve{\delta}$ is estimated, by solving the optimization problem

$$\min \rho^2(t), \text{under the constraint} \qquad \delta_1 \in [0, 0.1]\,\mathrm{s} \tag{4.16}$$

Matlab has been used for simulation and optimisation. The function *fminbnd* has been used for delays estimation. The algorithm is based on the golden section search and parabolic interpolation (Reklaitis et al., 1983). This function finds the minimum of

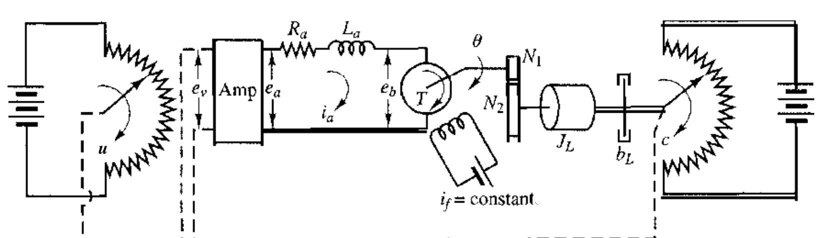


Figure 4.3: Schematic diagram of a control position for a DC motor.

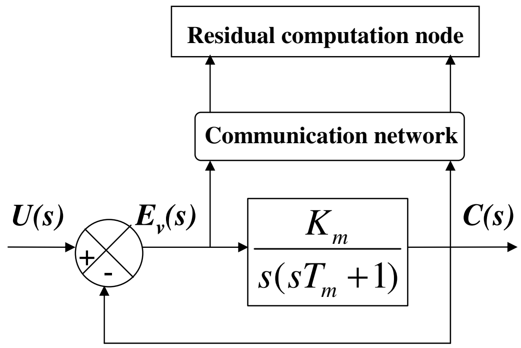


Figure 4.4: Block diagram of the system.

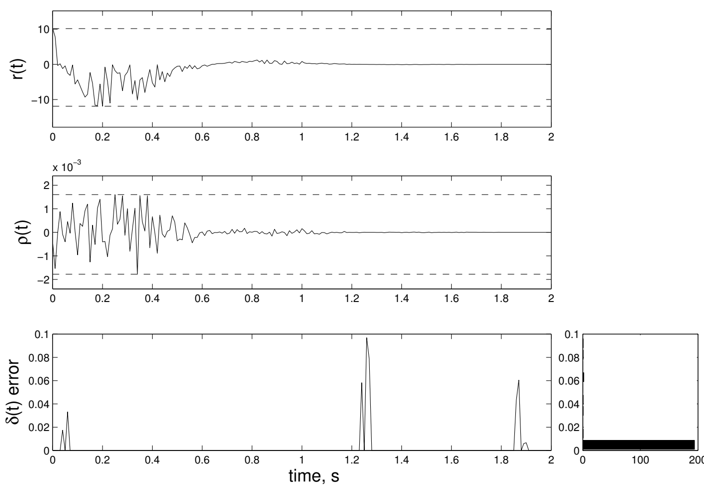


Figure 4.5: Residual in fault free situation and comparison of actual and estimated delays

a function within a fixed interval, the interval fixed for simulation was $[0, 0.1]$.

Fault free situation

Figure 4.5 depicts the residual computed without and with the delay estimation in a fault free situation ($r(t)$ above and $\rho(t)$ below). Notice that during the transient response, defined by the interval $[0, 1.03]$s, the residuals are more sensitive to the actual delays than during the steady-state time window. It is well known that false alarms can always be avoided (e.g. by never firing any alarm). This can be seen on Figure 4.5, which shows that false alarms are avoided in both cases, but at the cost of four orders of magnitude on the decision threshold (± 10 instead of $\pm 1.5 \text{x} 10^{-3}$). This means worse performances when faults will be present (larger missed detection rate, bigger size for faults to be detectable).

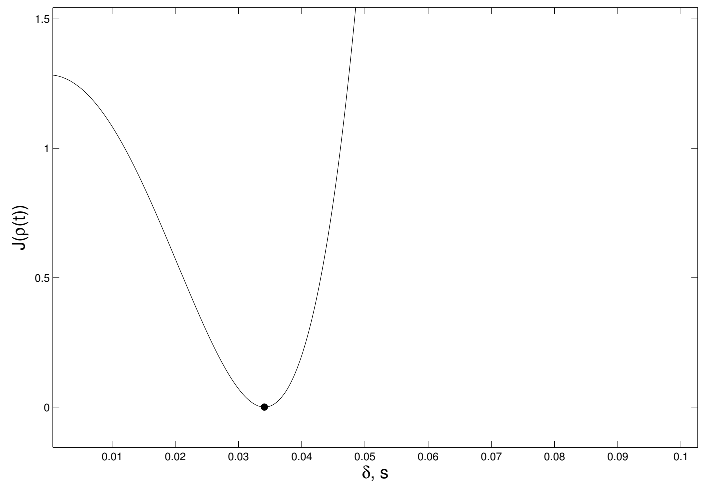


Figure 4.6: Cost function $J(\rho(t))$

The error distribution between actual and estimated delays is depicted in the histogram shown on Figure 4.5. The error is close to zero, which means a good estimation. The error increases during the steady-state time window because the excitation condition is not fulfilled, nevertheless it does not affect the computation of the residual $\rho(t)$ as can be seen on the figure.

Figure 4.6 depicts the quadratic cost function $J(\rho(t))$ to be minimized at one time instant, in that case the actual delay was 0.304s, the dot on the figure is the minimum of the cost function and therefore the estimated delay.

Faulty system situation

Figure 4.7, represents residuals $r(t)$ and $\rho(t)$ in the presence of an additive fault introduced on the sensor $e(t)$ as a constant 0.2 bias during the time window $[0.3, 0.6]$s. The effect of this fault on residual $r(t)$ is not large enough to overpass the thresholds, causing missed detections during all the fault instants. On the other hand, the lower figure depicts how $\rho(t)$ allows a proper detection. The error between actual and estimated delays increases significantly during the fault instants, but it does not affect the detection of the fault.

4.5.2 Considering delays in two signals

Let δ_1 and δ_2 be uniformly distributed in the interval $[0, 0.1]$ s.

Under $H^0_{system} \wedge H^0_{network}$ one obviously obtains $\rho\left(t\right) = 0$ when δ_i are replaced by their actual values. When these values are unknown, the admissible delays $\breve{\delta}_i$ are estimated, by

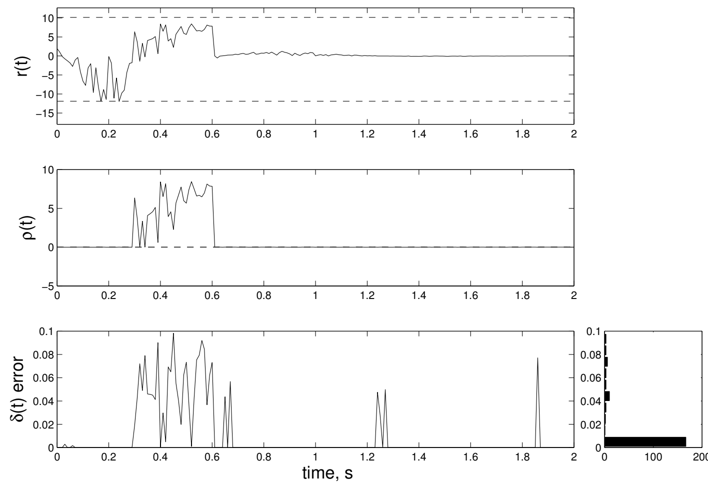


Figure 4.7: Residual in faulty situation and comparison of actual and estimated delays

solving the optimization problem

$$\min \rho^2(t), \text{under the constraint} \qquad \delta_i \in [0, 0.1]\,\mathrm{s} \tag{4.17}$$

Matlab has been used for simulation and optimisation. The function *fmincon* has been used for delays estimation. This function uses a sequential quadratic programming method (Reklaitis et al., 1983), and it is suitable for finding a minimum of a constrained nonlinear multivariable function. Initial condition, δ^0, for the algorithm has been fixed at zero.

Fault free situation

Figure 4.8 depicts the residual computed without and with the delay estimation in a fault free situation ($r(t)$ above and $\rho(t)$ below).

The comparisons of actual (δ_1 and δ_2) and estimated ($\breve{\delta}_1$ and $\breve{\delta}_2$) delays, and their distributions, are depicted in Figure 4.8. Due to the lack of a time reference it seems that estimations are not well achieved. In order to introduce a time reference, the estimation error is computed as the difference between the comparisons of the actual and estimated delays, see Figure 4.8. The estimation error is close to zero and it increases during the steady-state time window because the excitation condition is not fulfilled, nevertheless it does not affect the computation of the residual $\rho(t)$ as can be seen in Figure 4.8.

Figure 4.9 depicts the quadratic cost function $J(\rho(t))$ to be minimized at one time instant, in that case the actual delays were $\delta_1 = 0.044s$ and $\delta_2 = 0.055s$, the dot on the figure is the minimum of the cost function $J(\rho(t)) = 1.92\text{x}10^{-9}$.

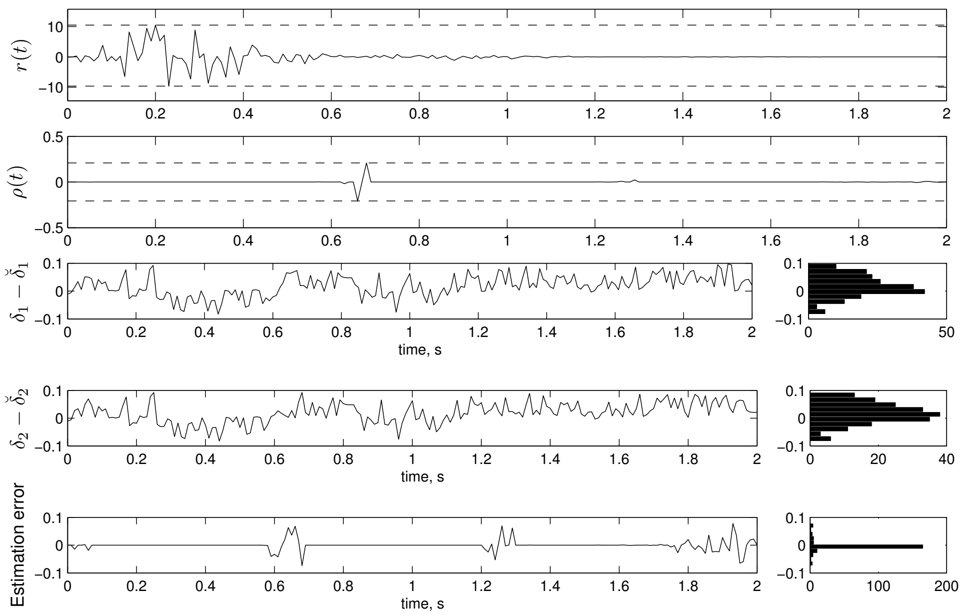


Figure 4.8: Residual in fault free situation, comparisons of the actual and estimated delays and the estimation error.

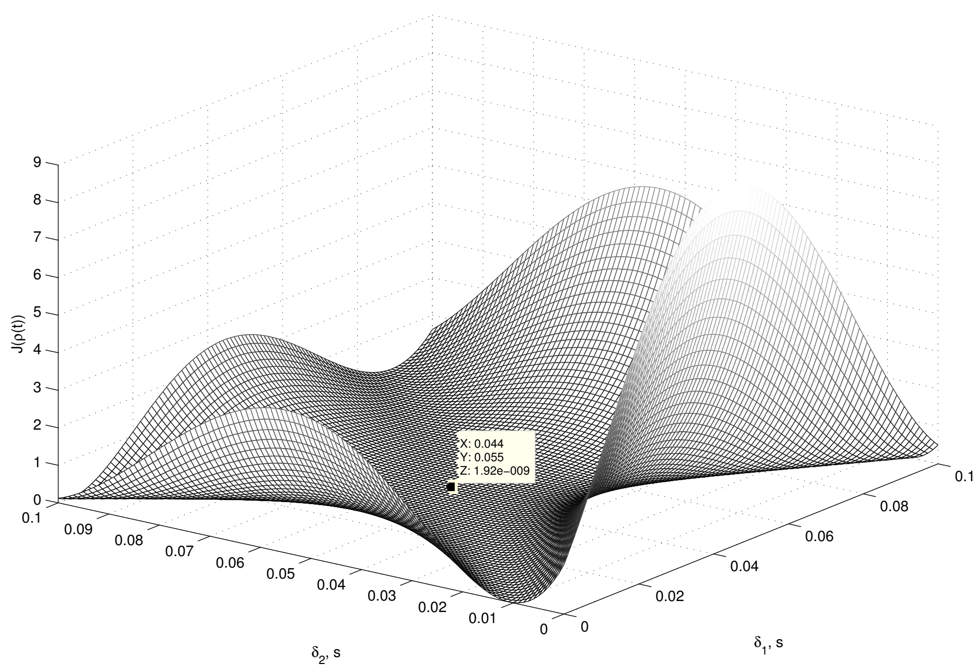


Figure 4.9: Cost function $J(\rho(t))$

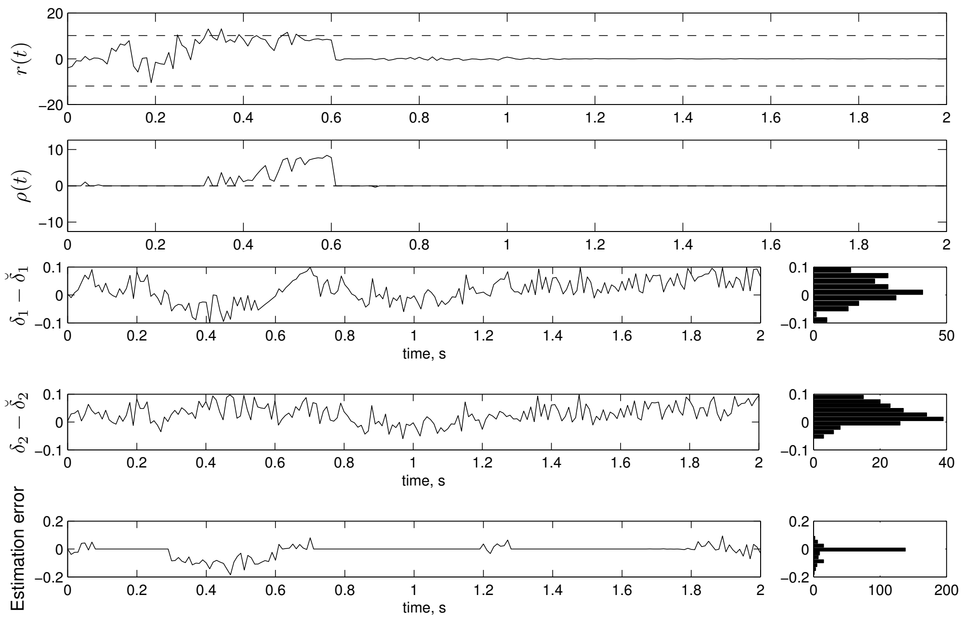


Figure 4.10: Residual in faulty situation, comparisons of the actual and estimated delays and the estimation error.

Faulty system situation

Figure 4.10, represents residuals $r(t)$ and $\rho(t)$ in the presence of an additive fault introduced on the sensor $e(t)$ as a constant 0.2 bias during the time window $[0.3, 0.6]$s. The effect of this fault on residual $r(t)$ is not large enough to overpass the thresholds, causing missed detections during almost all the fault instants. On the other hand, the lower figure depicts how $\rho(t)$ allows a proper detection. The estimation error increases significantly during the fault instants, but it does not affect the detection of the fault.

4.6 Conclusions

In model-based Fault Detection and Isolation a set of residuals is computed, that should be ideally zero in the fault-free case and different from zero, in the faulty case. However, in practice, residuals are different from zero, not only because of faults but because of measurement noise, unknown inputs modelling uncertainties and transmission delays in distributed systems. Using these residuals will produce false alarms and missed detections.

In this chapter it is shown that when transmission delays are known, it is possible to take them into account in the residual computation, thus introducing a delayed but otherwise unchanged decision and avoiding false alarms due to delays. However, when delays are unknown it is necessary to estimate them in order to compensate for their effect in the decision procedure. In this case, based on a very rough model of the delays, the chapter proposes to address the problem as an optimization problem. A search algorithm is used for the delay estimation by minimizing the residual under the constraints given by the transmission model. When the persistent excitation condition is fulfilled, the delays estimation can be carried out giving reliable results and avoiding false alarms.

The efficiency of the proposed approach is applied on a control position for a DC motor.

Chapter 5

Reduction of False Alarms in Fault Detection of Networked Control Systems with Data Dropout

5.1 Introduction

In NCS the availability of data is affected by different casuistics depending on network operation (traffic, polling strategies, hubs and switches delays, etc.) resulting in data dropout at a specific node where they are analysed. Thus, the residual computation may be inconsistent with the system state at the computation instant. Therefore, residuals that should theoretically be zero during the normal operation conditions could generate false alarms as the result of missing data or dropouts. This effect can be observed as an increase of false alarm ratios. A common solution in such situations is to increase the decision threshold according to the variability of the residual observed during data dropout. The cost of this action is a reduction of the sensitivity to faults and consequently an increase of missed detections.

In this Chapter, a formulation of how residual must be computed with available data is proposed. The aim is to reduce the false alarm ratio caused by data dropout without increasing the number of missed detections.

The chapter is organized as follows: Section 5.2 presents the influence of data dropout in residual computation. A technique for false alarms reduction due to data dropout is described in Section 5.3. An illustrative example in a laboratory plant is shown in Section 5.4. Finally some concluding remarks are given in Section 5.5.

5.2 The influence of data dropout in residual computation

Consider the discrete time deterministic system modelled by

$$x(k+1) = f(x(k), u(k), \varphi(k)) \tag{5.1}$$

$$y(k) = g(x(k), u(k), \varphi(k)) \tag{5.2}$$

where $x(k) \in R^n, u(k) \in R^r, y(k) \in R^m$ and $\varphi(k) \in R^q$ are respectively the state, control, output and fault vector, and f and g are given smooth vector fields.

Due to delays and possible dropout $z_i(k)$, generated at the production nodes, and $\hat{z}_i(k)$ data available at the residual computation node are distinguished and their dependence with the network communication status represented by:

$$\hat{z}_i(k) = \begin{cases} z_i(k) & if \quad \psi_i(k) = 0 \\ \hat{z}_i(k-1) & if \quad \psi_i(k) = 1 \end{cases} \tag{5.3}$$

where $\psi_i(k)$ is a stochastic variable which represents the data communication status. $\psi_i(k) = 0$ means that data at time k arrives correctly and therefore we are dealing with updated values $z_i(k)$, while $\psi_i(k) = 1$ means that data has not properly arrived and therefore the computation node only can use previous existent data $\hat{z}_i(k-1)$.

In general $\psi_i(k)$ should be modeled by means of probability distribution $\mathcal{P}_i$. Hence, the normal operation of the communication network on a given time window $[\alpha, \beta[$ can be described as a discrete deterministic model:

$$\begin{gathered} [H^0_{network} \implies \forall i \in I, \forall k \in [\alpha, \beta[: \psi_i(k) \sim \mathcal{P}_i] \\ \Longleftrightarrow \\ [\exists i \in I, \exists\, [\gamma, \lambda[\subseteq [\alpha, \beta[: \psi_i(k) \nsim \mathcal{P}_i \implies H^1_{network}] \end{gathered} \tag{5.4}$$

where $H^0_{network}$ represents the normal operation of the network in terms of $\mathcal{P}_i$ and $H^1_{network}$ is the hypothesis for anomalous behaviour.

5.3 Feasible residual computation

Residual computation computed as:

$$r(k) = G_c\left(z_i(k), i \in I\right) \tag{5.5}$$

can only be computed in case of availability $z_i(k) \forall i \in I$. Instead of this, only a residual $\hat{r}(k)$ can be computed with available data $\hat{z}_i(k)$, at instant k:

$$\hat{r}(k) = G_c\left(\hat{z}_i(k), i \in I\right) \tag{5.6}$$

and that will be the correct residual at k only if $\psi_i(k) = 0 \; \forall i \in I$:

$$\hat{r}(k) \begin{cases} = r(k) & if \quad \forall i \in I \quad \psi_i(k) = 0 \\ \neq r(k) & if \quad \exists i \in I \quad \psi_i(k) \neq 0 \end{cases} \tag{5.7}$$

In case of missing data, the only residual we can compute is with available data. Thus, under that circumstances we can use the last available residual, $\rho(k)$:

$$\rho(k) = \begin{cases} r(k) & if \quad \forall i \in I \quad \psi_i(k) = 0 \\ \rho(k-1) & if \quad \exists i \in I \quad \psi_i(k) \neq 0 \end{cases} \tag{5.8}$$

This feasible algorithm allows to compute the residual $\rho(k)$ equal to the residual $r(k)$ only if at the instant k all the measurements $z_i(k)$ are available. In case of missing data, some $z_i(k)$ will not be available and, then, the only possibility is to compute the residual $\rho(k)$ as the residual computed at the previous instant $\rho(k-1)$. That is, the available residual $\rho(k)$ will be equal to the residual computed at the last instant in which all the measurements were available.

$$\begin{gathered} \rho(k) = r(k-\tau) \quad if \quad \forall i \in I \quad \psi_i(k-k^{'}) = 0, \\ \tau = min(k^{'}) \quad : \quad \psi_i(k-k^{'}) = 0 \end{gathered} \tag{5.9}$$

where τ is the number of consecutive instants in which data missing occurs. If at the instant $k-\tau$ the data communication status $\psi_i(k-\tau) = 0 \quad \forall i \in I$, then

$$\rho(k) = r(k-\tau) = G_c(z_i(k-\tau)) = \hat{r}(k-\tau) \tag{5.10}$$

It is worth to notice that the goal is to compute the correct residual in presence of missing data instead of minimising its value as has been developed in Chapter 4.

5.3.1 The decision procedure under missing data

When the vector of data communication status

$$\psi(k) = (\psi_i(k), i \in I)$$

is perfectly known, the decision procedure:

$$\begin{gathered} [H^0_{system} \implies (\rho(k) = r(k-\tau)) \in \mathcal{N}(0)] \iff \\ [(\rho(k) = r(k-\tau)) \notin \mathcal{N}(0) \implies H^1_{system}] \end{gathered} \tag{5.11}$$

can be directly run by choosing

$$\tau = min(k^{'}) \quad : \quad \psi_i(k-k^{'}) = 0 \tag{5.12}$$

however when $\rho(k) \notin \mathcal{N}(0)$ the fault detection process is delayed by τ.

When $\psi(k)$ is unknown, the decision has to be done taking into account its behavior model. Therefore, from (5.4) and (5.11) the following decision logic is true:

$$\begin{gathered} [H^0_{system} \wedge H^0_{network}] \implies \exists \tau : [(\psi(k-\tau) = 0) \wedge \\ (\psi(k) \sim \mathcal{P}) \wedge ((\rho(k) = r(k-\tau)) \in \mathcal{N}(0))] \iff \\ \forall \tau : [(\psi(k-\tau) \neq 0) \vee (\psi(k) \nsim \mathcal{P}) \vee \\ ((\rho(k) = r(k-\tau)) \notin \mathcal{N}(0))] \implies [H^1_{system} \vee H^1_{network}] \end{gathered} \tag{5.13}$$

This decision logic expresses that the non existence of a τ such that (1) there is not missing data ($\psi(k-\tau)=0$), (2) the network is in normal operation $\psi(k) \sim \mathcal{P}$ and (3) the residual ($\rho(k) = r(k-\tau)$) lies inside $\mathcal{N}(0)$ evidences that the system, the network, or both do not operate properly. Even though both faults in the network and in the system can be detected, they cannot be isolated from each other in the absence of extra information.

Estimating the communication status

As we are dealing with raw data $\hat{z}_i$, the only thing we can do is to use them in order to determine the value of the communication status, $\psi(k)$. Techniques such as Data Reconciliation (Maquin and Ragot, 1991; Chouaib, 2004) and Sensor Fusion algorithms (Z. Bak et al., 1998; Blanke, 2005) have been used in order to deal with delayed measurements; but this techniques use the mathematical model of the system to get their purpose. Since the idea is to measure the communication status as soon as possible, using mathematical model of the system may increase the computation time to carry out this task and therefore on the fault detection procedure.

The next simple and feasible algorithm is proposed for measuring the communication status, $\psi(k)$:

$$\psi(k) = \begin{cases} 0 & if \quad \forall i \in I \quad \hat{z}_i(k) - \hat{z}_i(k-1) \neq 0 \\ 1 & if \quad \exists i \in I \quad \hat{z}_i(k) - \hat{z}_i(k-1) = 0 \end{cases} \tag{5.14}$$

The algorithm is based on the comparison of the available data, $z_i(k)$, with its value on the previous time instant $z_i(k-1)$. If all the comparison are different to zero, it is assumed that there is not a data dropout. But if there is one comparison equal to zero, therefore it is assumed that there has been a data dropout.

It should be emphasized that the decision rule 5.14 obviously does not take into account the fact that constant data during steady state operation are not associated with data dropout. However, this does not affect the computation of the residual $\rho(k)$ as it will be appreciated in the illustrative example.

5.4 Illustrative example: Fault detection in a NCS laboratory plant

5.4.1 Description of the system

The proposed approach has been used on a level control laboratory plant instrumented with a networked control and supervisory system. In this plant (See Figure 5.1) a pump sends liquid from one tank to another one that is placed in a higher level. Water returns to the first tank by the effect of gravity. A PID control maintains the level in the higher tank by governing the pump.

The system has been instrumented with sensors (flow-meter and level sensor connected to a SIEMENS ET200U module) and a PID controller (BÜRKET 1100) accessible both via PROFIBUS as independent nodes. A FDI (Fault Detection and Isolation) module

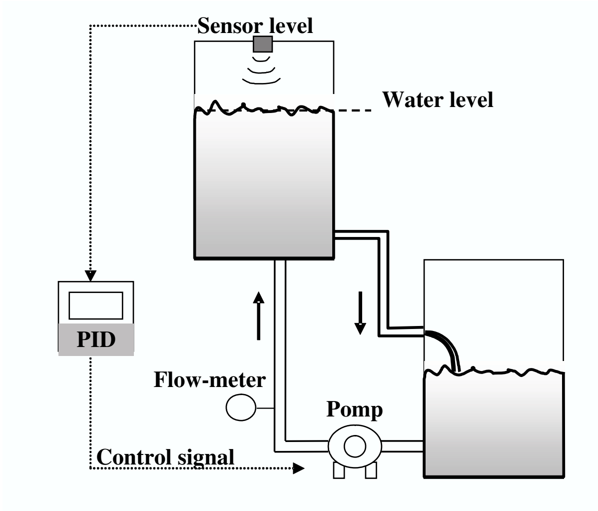


Figure 5.1: Laboratory plant

has been implemented in a SCADA (Supervisory Control and Data Acquisition) system running in a computer accessing data from an OPC server connected to the PROFIBUS in a master/slave scheme as depicted in Figure 6.3. Supervisory system can be either in the local computer, directly connected to the PROFIBUS, or in a remote computer connected to the OPC server via ethernet.

The FDI system has been designed based on Analytical Redundancy Relations extracted from the following model of the system:

$$c_1 : Q_i = u.\bar{Q}_i \tag{5.15}$$

$$c_2 : Q_o = K.\sqrt{h} \tag{5.16}$$

$$d_3 : \dot{h} = \frac{d}{dt}h \tag{5.17}$$

$$c_4 : \dot{h} = \frac{1}{A}(Q_i - Q_o) \tag{5.18}$$

$$m_1 : h = h_{m_1} \tag{5.19}$$

$$m_2 : Q_i = Q_{i,m_2} \tag{5.20}$$

where Q_i, h and u are monitored process variables that represent the input flow, the level of the higher tank and the control signal, respectively. Q_o is the output flow of the higher tank. $\bar{Q}_i$ is a parametrization of the pump. A and K are known parameters relative to tanks surface and pressure in the pipes. And m_1 and m_2 are additional measurement constraints given the a flow-meter and sensor of level.

Applying the structural analysis technique (Blanke et al., 2003) the following computational forms for the residuals in discrete time are obtained (See details in Appendix B):

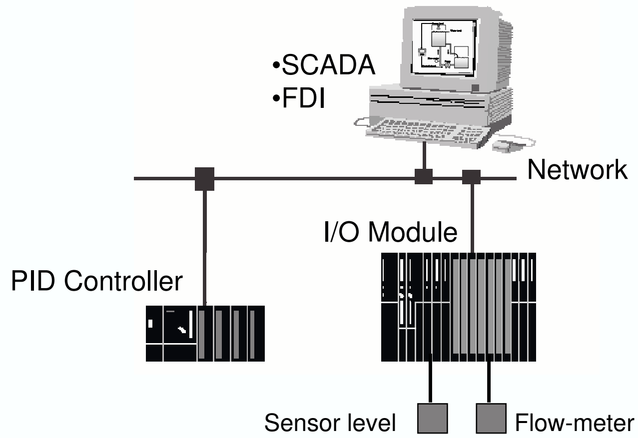


Figure 5.2: Process communication

$$r_1(k) = Q_i(k) - K\sqrt{h(k)} - A(h(k) - h(k-1)) = 0 \tag{5.21}$$

$$r_2(k) = u(k).\bar{Q}_i - Q_i(k) = 0 \tag{5.22}$$

Since the data available at the residual computation node are identified by $\hat{u}(k)$,$\hat{h}(k)$ and $\hat{Q}_i(k)$, the previous equations can be rewritten to show the equations for the vector of available residual at the computation node when affected by the data dropout, $\rho(k)$:

$$\rho_1(k) = \hat{Q}_i(k) - K\sqrt{\hat{h}(k)} - A(\hat{h}(k) - \hat{h}(k-1)) = 0 \tag{5.23}$$

$$\rho_2(k) = \hat{u}(k).\bar{Q}_i - \hat{Q}_i(k) = 0 \tag{5.24}$$

Remember that although equation forms are the same, meaning is different. $r(k)$ and $\rho(k)$ only coincide if data available at instant k corresponds exactly with measurements at this instant.

5.4.2 Dropout communication model

For testing purposes the whole supervisory system (SCADA and FDI) has been implemented in the same computer and a simple module has been added to simulate communication delays and dropout in the reception of data for FDI purposes. This allows modelling random network delays of a variety of communication systems and improve the repeatability of results for research purposes. For this example random network delays have been modelled by a Poisson process and it has been considered that a data dropout occurs when delays exceed a predefined threshold. The Poisson distribution, (5.25), is representative of random network delays introduced by queuing at routers and frame collision on the ethernet networks and the propagation time delays from the PROFIBUS network (Tipsuwan and Chow, 2003). Since the residual is computed periodically at a fixed sample time, the dropout threshold has been fixed to be equal to this sample time. This simple model is enough to demonstrate the benefits of the proposed approach.

$$\mathcal{P}(\lambda, \theta) = \lambda^{\theta} \frac{e^{-\lambda}}{\theta!} \tag{5.25}$$

Thus, data dropout is modelled by a Poisson distribution (5.25) expressing the probability of occurring a number of data dropouts, θ, in a fixed period of time (time window needed for residual computation) when these events occur with a known intensity or average rate represented by the mean delay, λ, and under the assumption of independence among the occurrence of these dropouts. The value of λ used in the experiments has been estimated from the real system as 0.7. This is an average representative value obtained with normal traffic load and forcing the cycling service in the PROFIBUS network to 1 second. The same sampling time has been used for the residual computation.

5.4.3 Normal Operation

Several scenarios have been analysed to show the benefits of maintaining the last computed residual when data dropout occurs instead of using delayed data. The first scenario represents the normal operation conditions. In the second one data dropout is introduced to the level sensor and its influence in fault detection is evaluated. Finally the presence of dropouts in two sensors has been considered. Performance has been measured in terms of false alarms ratios and missed detections.

During normal operation conditions false alarms occurs in transients provoked by sudden set point variations (step at t=40 s) due to the presence of time shifted variables involved in the residuals. Figure 5.3 depicts the classical residuals r_1 and r_2, the tank level $\hat{h}$ and the input flow $\hat{Q}$ during normal situation. Decision thresholds have been determined in this scenario as $\bar{x} \pm 3\sigma$, where $\bar{x}$ is the mean and σ is the standard deviation. These fixed values have been used in the next scenarios.

Data dropout in one sensor measurement

Figure 5.4 represents the classical residuals, (r_1 and r_2), affected by data dropout on the level sensor without faults. The vertical dotted lines, between the thresholds, represent the time instances when the data are dropped out. The ratio of false alarms, $FAR_T(\%)$, has increased during the transient and it has been calculated following the expression:

$$FAR_T(\%) = \frac{FA_T}{Total_T} \times 100 \tag{5.26}$$

where $Total_T$ is the number of evaluations of the residual performed during a time window $T = [T_{ini}, T_{end}]$ and FA_T is the number of these evaluations resulting in false

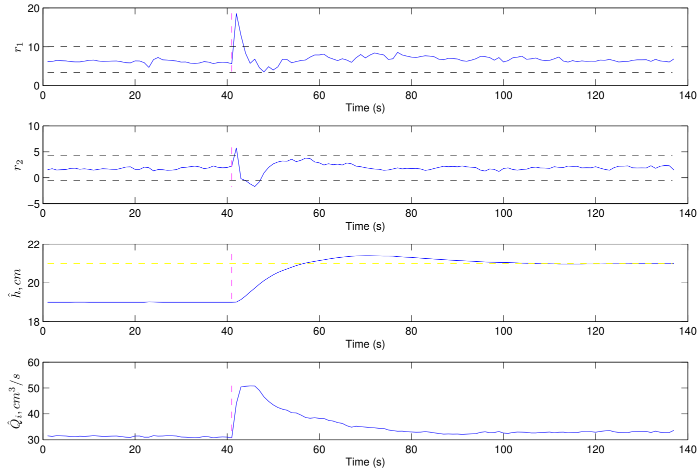


Figure 5.3: Residuals r_1 and r_2, tank level and input flow during normal situation.

	FAR_T		FAR_T
r_1	22.5%	ρ_1	4%
r_2	12.5%	ρ_2	2.5%

Table 5.1: False Alarm Ratio in $T = [40, 80]$s, when dropout occurs in one sensor

alarm. In the example the time window has been fixed to $T = [40, 80]$ s. In order to consider only the transient behaviour of the system, resulting for r_1, the residual affected by the level sensor, a $FAR_T = 20\%$. However, when the available residual (ρ_1) is used with the same decision thresholds (See Figure 5.5), the false alarm ratio evaluated in the same time window is reduced to 7.5%.

Usually there exist a trade off between false alarms and missed detections. Missed detection ratio, $MDR_T(\%)$, can be computed analogously to equation (5.26) by simply substituting the number of false alarms by the number of missed detection in the evaluation time window. This aspect has been analysed during the presence of an additive fault in the flow-meter (a bias of 2.3% from nominal value) has been introduced during the time window $T = [66, 99]$ s.) as it can be observed in Figure 5.6. False alarms produced by the data dropout during the transient, induced by the set point change, avoids the perfect detection of the fault when it appears. Instead of this, when the available residuals (ρ_1 and ρ_2) (Figure 5.7) are used, false alarms provoked by dropouts are eliminated at same time that the missed detection ratio maintains practically the same value close to zero. In fact, additive faults in sensors are not affected by system dynamics and consequently only small variations on its detection has to be observed in both residuals.

Data dropout in two sensor measurements

In order to test the performance of the proposed method in presence of a bigger number of dropouts, this effect has been simulated in two sensors simultaneously: level and flow

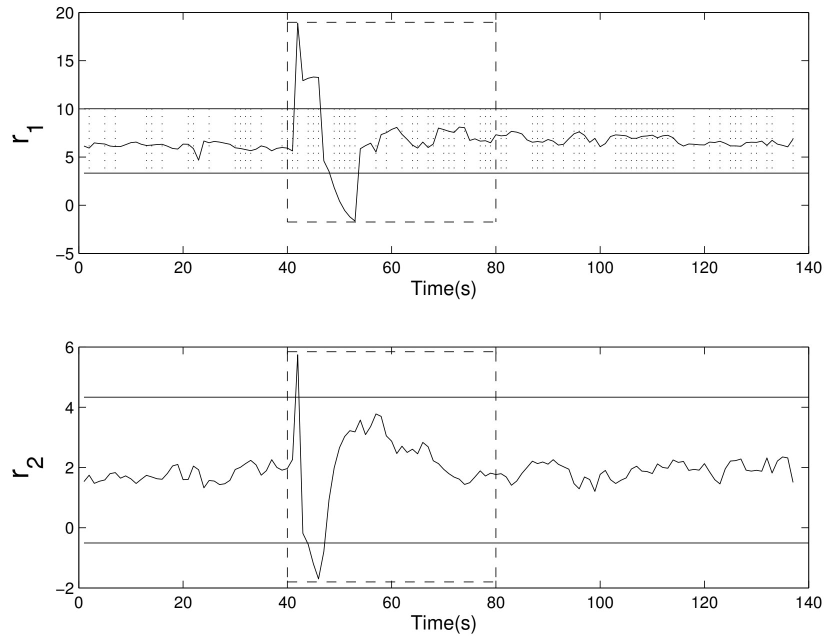


Figure 5.4: Residuals r_1 and r_2 affected by data dropout on the sensor level.

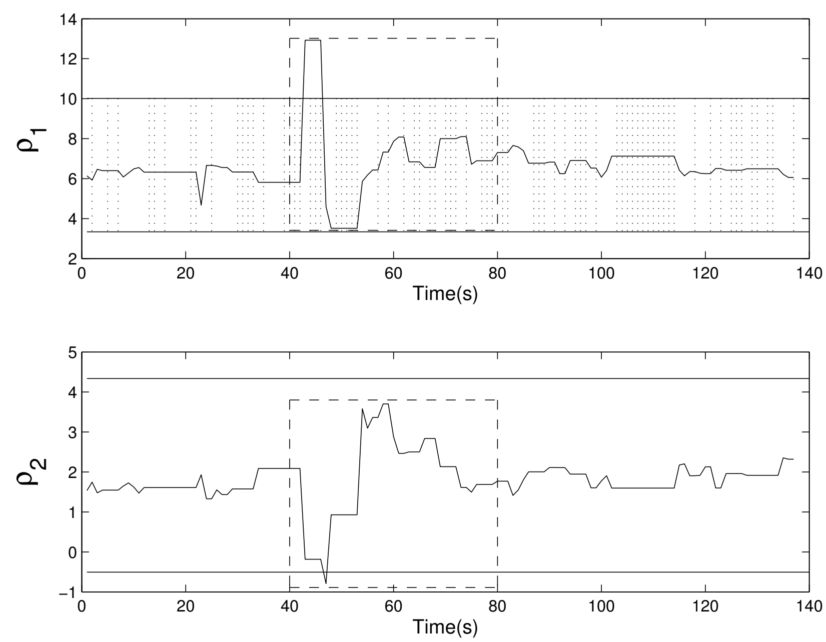


Figure 5.5: Residuals ρ_1 and ρ_2 in presence of data dropout on the sensor level.

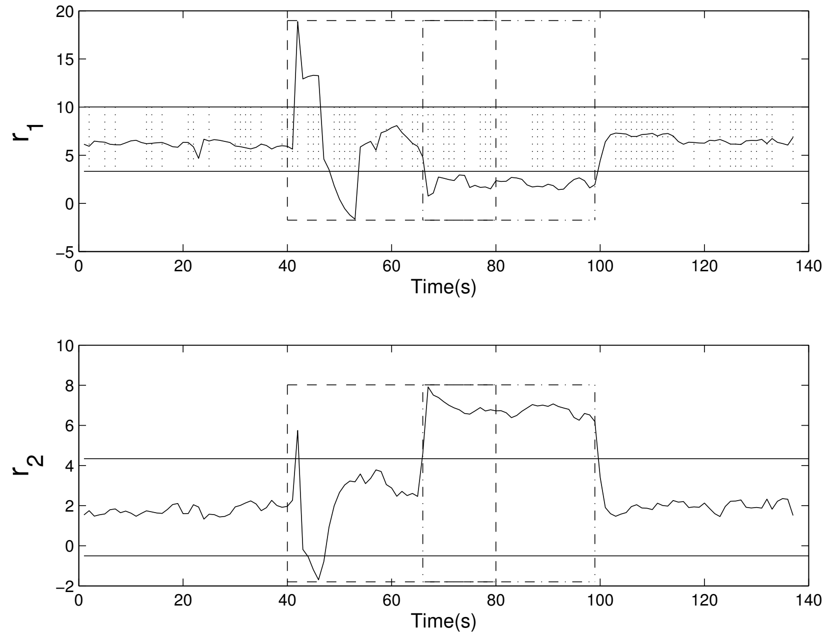


Figure 5.6: Residuals r_1 and r_2 affected by data dropout on the sensor level and a fault on the flow-meter.

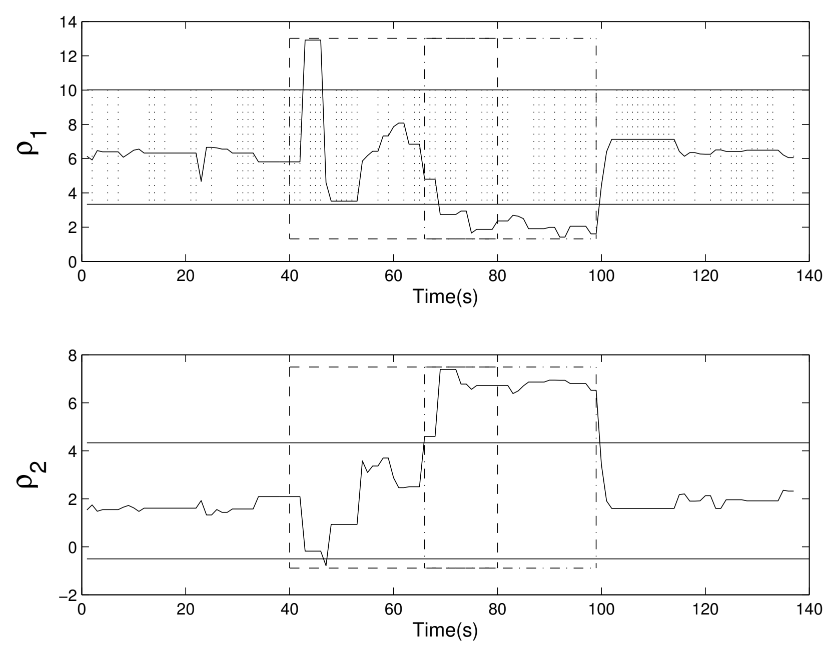


Figure 5.7: Residuals ρ_1 and ρ_2 during data dropout on the sensor level and a fault on the flow-meter

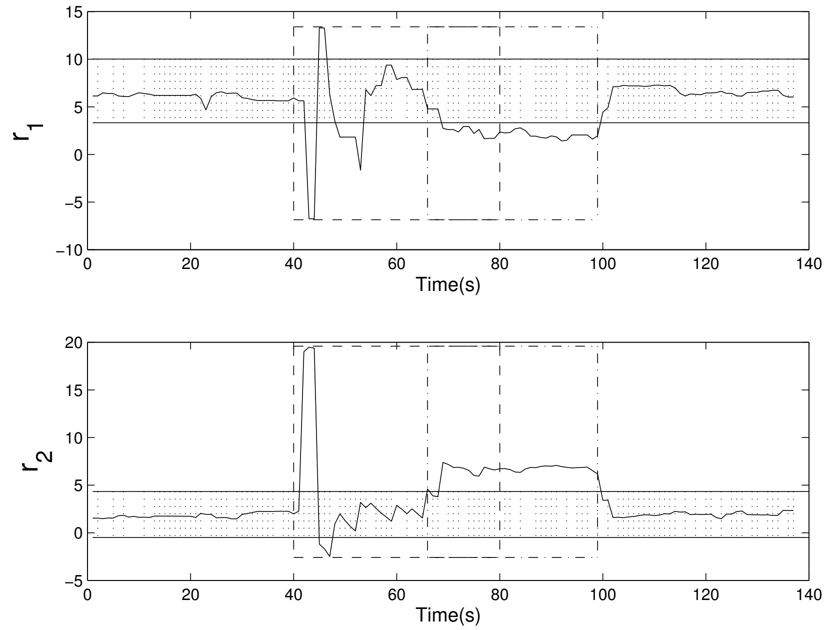


Figure 5.8: Residuals r_1 and r_2 in presence of fault in the flow-meter and data dropout in two sensors.

sensors. A similar effect could be obtained if the λ parameter, in the Poisson distribution, had been increased. The performance of the use of the available residual instead of the classical one has been tested in presence and absence of fault in the flow-meter.

Figures 5.8 and 5.9 show the respective behaviours of residuals r_i and ρ_i in the faulty situation when measurements from both sensors are affected by data dropouts. The effect of an increase of data dropout is evident in both figures. Evaluation the False alarm ratio in the same time window, $T = [40, 80]$s, this increase can be quantified and compared with respect the previous scenario where only one sensor was affected by data dropout. Also the benefits of using only available residuals for the detection instead of classical ones is evident from the ratios in table 5.2.

	FAR_T		FAR_T
r_1	27.5%	ρ_1	0%
r_2	15%	ρ_2	0%

Table 5.2: False Alarm Ratio evaluated in $T = [40, 80]$s, when dropout occurs in two sensors

Similarly to the previous situation the number of missed detection is not affected by these dropouts.

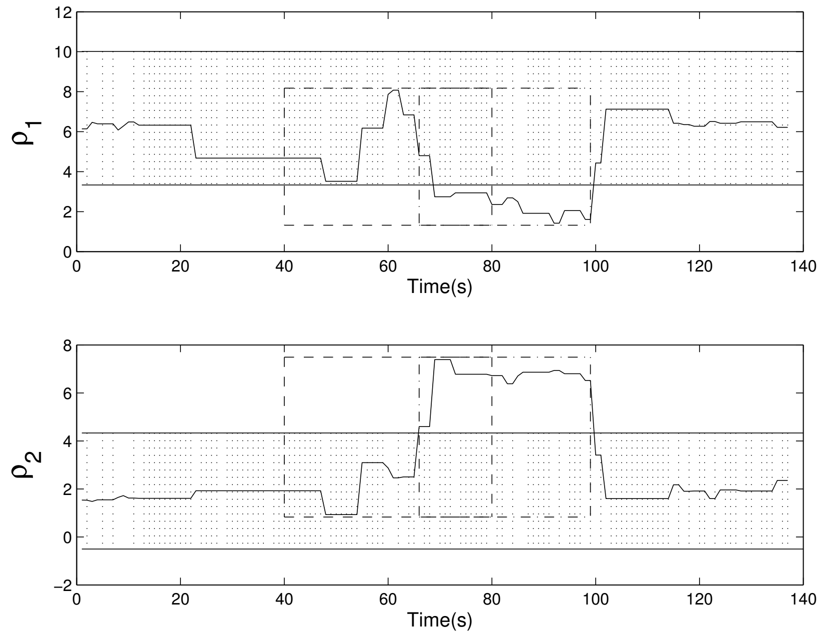


Figure 5.9: Residuals ρ_1 and $r\rho_2$ in presence of fault in the flow-meter and data dropout in two sensors.

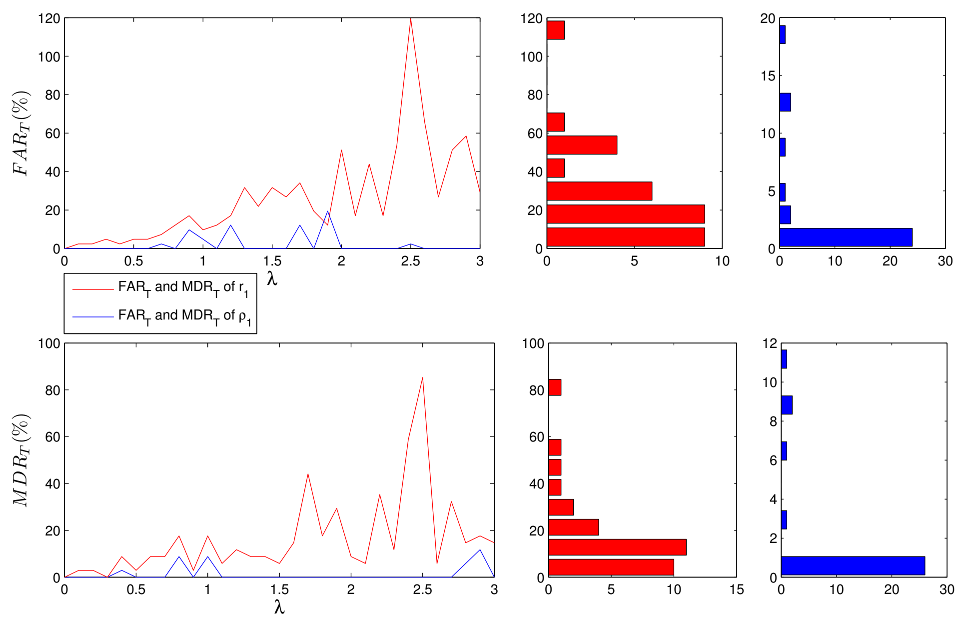


Figure 5.10: $FAR_T(\%)$ and the $MDR_T(\%)$ of residuals $r_1(t)$ and $\rho_1(t)$ versus the dropout rate.

Figure 5.10 depicts the False Alarm Ratio and the Missed Detection Ratio versus the dropout rate, ($FAR_T(\%)$ above and $MDR_T(\%)$ below). Dropout rate is obtained by means of the variation of the parameter λ in the Poisson distribution. The value of λ used in the previous experiments, with normal traffic load, has been estimated from the real system as 0.7, therefore the variation of this parameter has been done within the interval $[0, 3]$. The $FAR_T(\%)$ of residuals $r_1(t)$ and $\rho_1(t)$, painted in Figure 5.10 in red and blue respectively, have been computed under normal operation conditions and data dropouts on the sensor level. The $MDR_T(\%)$ of residuals $r_1(t)$ and $\rho_1(t)$, painted in red and blue respectively, have been computed under the presence of an additive fault in the flow-meter (a bias of 2.3% from nominal value) and data dropouts on the sensor level.

Notice that residual r_1 is very sensitive to the augment of data dropout, false alarms and missed detection increase considerably as data dropout augment. While the residual ρ_1 evidences a high efficiency because it allows to reduce the false alarms without a considerable increasing of missed detections.

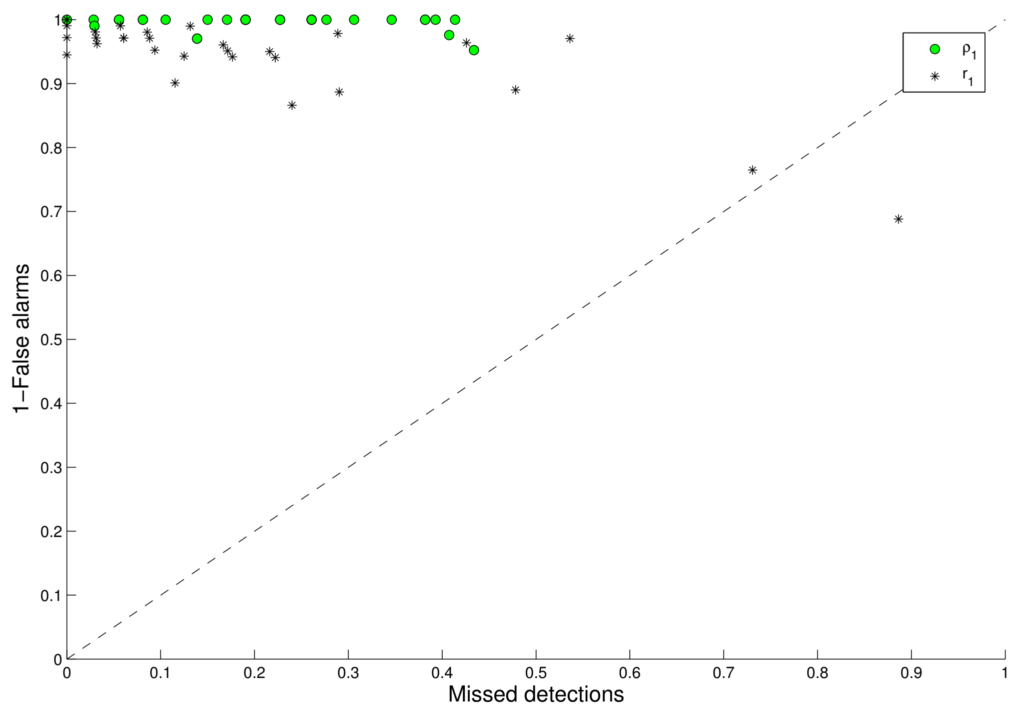


Figure 5.11: Relation between false alarms and missed detections as the λ parameter is varied.

Figure 5.11 depicts the relation between false alarms and missed detections as the λ parameter of the Poisson distribution is varied. In the figure, the asterisks and the green circles represent the result with residuals r_1 and ρ_1, respectively. Notice that most of the green points are located around the top left cite of the figure which means few false alarms and few missed detection, while several asterisks evidence more false alarms and more missed detection than the green points.

5.5 Conclusions

A simple modification for residual computation has been proposed in this chapter in order to reduce the effect of missing values at the time of residual computation. This dropout is the consequence of different delays in the reception of data when working with networked systems. The most evident effect of data droput in the fault detection system is the increase of false alarm ratios. A simple formulation has been proposed to solve this problem based on the computation of the residual with the available data at each time instant instead of using the classical formulation. Principal benefit is the drastic reduction of false alarm ratio avoiding the classical trade off with missed detection. The efficiency of the proposed approach is illustrated on a laboratory plant, where the false alarms during the transient time window were reduced without increasing the missed detections under different conditions of data dropout.

Chapter 6

Dynamic Time Warping for Residual Computation

6.1 Introduction

There are numerous studies applied to time series that have been carried out in order to compare and classify similar patterns by means of a similarity measure. Most algorithms that operate with data time series use the Euclidean distance or some variant. However, Euclidean distance could produce an incorrect similarity measure because it is very sensitive to time misalignment. Dynamic Time Warping (DTW) tries to solve this inconvenient. It uses dynamic programming (Sakoe and Chiba, 1978; Silverman and Morgan, 1990) to align time series with a given template so that the total distance measure is minimized.

In distributed control systems, the residual computation form is implemented as an algorithm in one node of the network. As has been studied in previous chapters, residual computation may be affected by time misalignment caused by communication delays or even by modelling errors, this will often generate false alarms. Indeed, even under normal operation, the time misalignment associated with the variables that appear in the computation form produce discrepancies of the residuals from zero.

During faulty situation the system model is affected, producing a discrepancy of the residual to zero because the consistency between the measurements and system model is not fulfilled. However, when the discrepancy is produced by communication delays it could be compensated by means of the correct alignment between the measurements. This chapter proposes the use of DTW for reduce the effects of time misalignment when residual computation is performed, (Llanos et al., 2004a; Gamero et al., 2004).

This chapter is organized as follows: Section 6.2 summarises the DTW algorithm. In Section 6.3, a modification of DTW in order to be applied on-lie is explained. In Section 6.4 the use of DTW for improving the residual computation in presence of time misalignment is formulated. Section 6.5 presents the applications in a laboratory plant. Finally some concluding remarks are given in Section 6.6.

6.2 Dynamic Time Warping

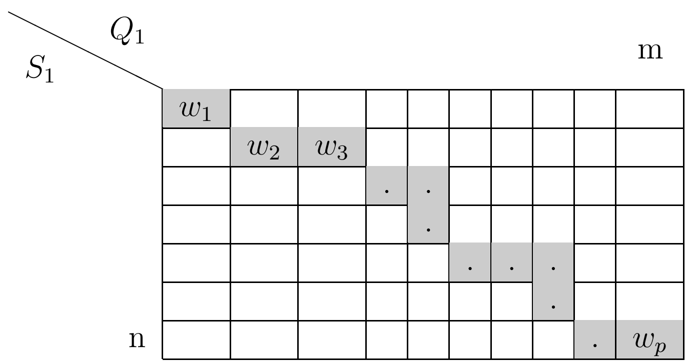
Dynamic Time Warping (DTW) is a technique that finds the optimal alignment between two time-series if one time-series may be "warped" non-linearly by stretching or shrinking it along its time axis. This warping between two time-series can then be used to find corresponding regions between the two time-series or to determine the similarity between the two time-series.

Let S and Q denote two time-sampled signals

$$S = [s_1, s_2, \ldots, s_{n-1}, s_n] \tag{6.1}$$
$$Q = [q_1, q_2, \ldots, q_{m-1}, q_m] \tag{6.2}$$

a m-by-n grid can be developed, as illustrated in Table 6.1. Each grid element, (i, j), represents an alignment between points s_i and q_j. A *warping path* W is a sequence of grid elements that define as alignment between S and Q.

Table 6.1: A warping path in a m-by-n grid.



$$W = (i_1, j_1), (i_2, j_2), \ldots, (i_p, j_p) \tag{6.3}$$
$$max(n, m) \leq p < m + n - 1 \tag{6.4}$$

, where (i_k, j_k) corresponds to the k^{th} grid element in the warping path. For example, (i_3, j_3) in Table 6.1 represents the grid element $(2, 3)$, which implies that s_2 is aligned with q_3. For practical reasons, several types of constraints, which concern the warping path, are introduced in prevalent research works (Sankoff and Kruskal, 1983; Rabiner and Juang, 1993; Berndt and Clifford, 1994):

- **End point constraints**: the warping path should start at $(1, 1)$ and end at (n, m).
- **Monotonicity and Continuity**: given two grid elements in a warping path, (i_k, j_k) and (i_{k+1}, j_{k+1}), then $1 \geq i_{k+1} - i_k \geq 0$ and $1 \geq j_{k+1} - j_k \geq 0$. This restricts the allowable transitions of a node to adjacent elements, which located at either east, south, or south-east with respect to Table 6.1.

- **Global Path Constraint**: the global path constraint defines the region of grid elements that are searched for optimal warping path. The warping path is limited within the warping (Berndt and Clifford, 1994), which is known as Sakoe-Chiba Band. For example, the grey are in the Table 6.2 refers to such a band. The constraint can be defined as follows:

$$\forall (i_k, j_k) \in W, \quad i_k - r \leq j_k \leq i_k + r, \tag{6.5}$$

  where r is the width of the warping window. For example, in Table 6.2, $r = 2$.

Table 6.2: The warping path is restricted in the grey area (the warping window).

	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}
s_1	↖	·	·							
s_2	·	←	·	·						
s_3	·	·	↖	·	·					
s_4		·	·	←	·	·				
s_5			·	·	↖	·	·			
s_6				·	·	↖	←	·		
s_7					·	·	·	↑	·	
s_8						·	·	↑	·	·
s_9							·	·	↖	·
s_{10}								·	·	↖

After aligning the sequences S and Q, their similarity can be measured by the cumulative distance of the warping path between them. Each element in the warping path is associated with the distance, i.e. $d(i_k, j_k) = |s_{i_k} - q_{j_k}|$. The cumulative distance of a warping path e.g. $W = (i_1, j_1), (i_2, j_2), \ldots, (i_p, j_p)$, is defined as follows:

$$D_c = \sum_{k=1}^{p} d(i_k, j_k) \tag{6.6}$$

There are possibility many warping paths. Among all the potential warping paths, an *optima warping path* can be chosen such that its cumulative distance, D_c, is minimum. The corresponding distance is defined as the *time warping distance*, D_{tw}.

$$D_{tw}(S, Q) = \min_{\forall W} D_c(W) \tag{6.7}$$

As there are many warping paths, searching through all of them is computational expensive. Therefore, dynamic programming approach (Rabiner and Juang, 1993; Yi et al., 1998) is proposed to find the *optimal warping path*. The approach is based on the following recurrence formula that defines the cumulative distance, $\gamma(i, j)$, for each grid element.

$$\gamma(i, j) = d(i, j) + \min \gamma(i-1, j), \gamma(i, j-1), \gamma(i-1, j-1) \tag{6.8}$$

Illustrative example

Consider two sequences:

$$S = [2, 5, 2, 5, 2, 1] \tag{6.9}$$
$$Q = [0, 3, 6, 0, 6, 0] \tag{6.10}$$

Table 6.3: A cumulative distance matrix for sequences Q and S.

S \ Q	**0**	**3**	**6**	**0**	**6**	**0**
2	2	3	7	9	13	15
5	7	4	4	9	10	15
2	9	5	8	6	10	12
5	14	7	6	11	7	12
2	16	8	10	8	11	9
1	17	10	13	9	13	10

by applying the dynamic programming algorithm, we can construct a cumulative distance matrix as shown in Table 6.3. Each value in the cell represents the cumulative distance of that cell. The cumulative distance of a cell is the sum of the distance associated with it and the minimum of the cumulative distances of its neighboring cells. For example:

$$\gamma(5,3) = d(4,3) + \min \gamma(3,3), \gamma(4,2), \gamma(3,2) = 6 \tag{6.11}$$

After filling up the table, the optimal warping path can be found by tracing backward from the lower right corner, $(6,6)$, towards the upper left corner, $(1,1)$. At each cell, we can choose the previous neighboring cell with minimum cumulative distance.

In the literature review on DTW applications for fault diagnosis we can mention the work done by (Kassidas et al., 1998a) where they used DTW to synchronise batch trajectories by combining it with multiway principal component analysis/partial lest squares (PCA/PLS). In (Kassidas et al., 1998b) they reported its use for fault detection and diagnosis in continuous chemical process, and (Nomikos and MacGregor, 1994) have reported its use for batch process monitoring.(Li et al., 2004) combined DTW with wavelet decomposition for synchronising batch trajectories. The original signals were decomposed into approximations and details at different scales and matched at each scale separately, using DTW. The matched signals were used, rather that the reconstructed signals, to obtain the syncronised signal.

In situations with differences in the magnitude of the two signals, DTW would try to solve the variability in the the Y-axis by warping the X-axis and, thus, result in inappropriate warping. To overcome this limitation, (Colomer et al., 2002) combined DTW with qualitative representation of signals. Each signal was first decomposed into episodes, which provided a higher-level representation of the signal. DTW was then used to find the optimal match between the episodes of the two signals, and tested for diagnosis of faults in a level control system.

(Srinivasan et al., 2004) proposed an alternative approach that constraints the search for the corresponding points of the two signals, based on landmarks in the signal that are derived from the perspectives of the operators. This constraints were used with DTW.

Above approaches are normally used for off-line applications due to computational expensive cost (in both time an memory) of DTW algorithm. In this thesis we propose a slight modification of DTW in order to adapt it for on-line application. Next section is devoted to explain the main particularities of the new algorithm.

6.3 On-line Dynamic Time Warping

Since DTW is a good method to compensate temporal distortions due to communication delays, this thesis proposes a modification of the algorithm in order to adapt it for on-line application. The new algorithm takes advantage of the *warping window* in order to use it as a sliding window. As main particularities, the two sequences have got the same length and the new algorithm returns a distance value at every sample time. So, it is necessary to obtain a finite sequence from original data to calculate new distances.

The algorithm starts at time $t = 2$ calculating the cumulative distance, $\gamma(i, j)$ for each grid element of the squared matrix. Later on, the matrix grows up and only cumulative distances for new cells in the matrix are calculated. Next, the matrix reaches a maximum value established according to the warping window width, r, and it becomes a sliding window. At each sample time oldest cells in the matrix are deleted and cumulative distances are calculated for empty cells corresponding to the new sample (figure 6.1). A new path must be found for each window and the distance value is obtained calculating the total distance according to this new path.

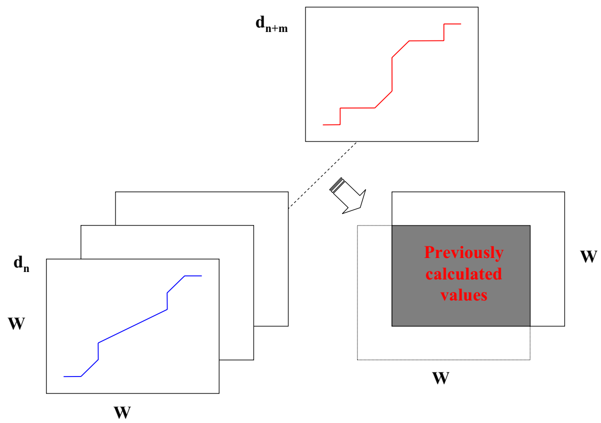


Figure 6.1: On-line DTW.

Illustrative example

Continuing with the example of the previous section 6.2, let's apply $DTW_{on-line}$ to sequences S and Q, with the width of the warping windows $r = 2$. Table 6.4 shows the cumulative distance at sample time $t = 3$, that is when the matrix reaches a maximum value established according to the warping window width, $r = 2$, and it becomes a sliding window.

Table 6.5 shows the cumulative distance matrix at sample time $t = 4$, grids colored in red denotes cumulative distances computed previously and which not have to be computed again, oldest cells (computed at $t = 1$) are not taken into account and cumulative distances are calculated just for the new values.

Table 6.4: A cumulative distance matrix for sequences Q and S at the sample time $t = 3$.

S \ Q	**0**	**3**	**6**
2	2	3	7
5	7	4	4
2	9	5	8

Table 6.5: Cumulative distance values of the sliding window for sequences Q and S at time $t = 4$.

S \ Q		**3**	**6**	**0**
5		4	4	9
2		5	8	6
5		7	6	11

Tables 6.6 and 6.7 show the cumulative distance values of the sliding windows at time $t = 5$ and $t = 6$, respectively.

Table 6.6: Cumulative distance values of the sliding window for sequences Q and S at time $t = 5$.

S \ Q			**6**	**0**	**6**
2			8	6	10
5			6	11	7
2			10	8	11

Table 6.7: Cumulative distance values of the sliding window for sequences Q and S at time $t = 6$.

S \ Q				**0**	**6**	**0**
5				11	7	12
2				8	11	9
1				9	13	10

Notice that the cumulative distance $\gamma(6, 6) = 10$ is the same distance value as applying traditional DTW algorithm. Nevertheless, both methods can not be compared because DTW needs all the sequences S and Q to find an alignment between them, while $DTW_{on-line}$ uses subsequences of S and Q to find an alignment within the warping window. In any case this fact is a way to confirm that the optimization is going in the right direction.

6.4 Dynamic Time Warping for improving Residual Computation in presence of time misalignments

As was stated in section 3.5.1, in distributed control systems, the residual computation form is implemented as an algorithm in one node of the network.

$$r(t) = G_c(\hat{z}_i(t), i \in I) \tag{6.12}$$

where $\hat{z}_i(t)$ is the data available at the residual computation node. Equation (6.12) can be transformed into an equivalent system:

$$r(t) = G_c(\hat{z}_i(t)) = G_a(\hat{z}_i(t)) - G_b(\hat{z}_i(t)) \tag{6.13}$$

and since $DTW_{on-line}$ can be applied to a couple of signals a new residual $\rho(t)$ less sensitive to time misalignments may be computed as follows:

$$\rho(t) = \mathbf{DTW_{on\text{-}line}}[G_a(\hat{z}_i(t)), G_b(\hat{z}_i(t))] \tag{6.14}$$

A particular case of this solution is the residual computation from observer based technique, where the system output is compared to estimated output. In this case $G_a = y(t)$ and $G_b = y^*(t)$, where $y(t)$ and $y^*(t)$ are the real and the estimated output of the system, respectively.

Illustrative example

Let's consider a residual:

$$r(t) = a\hat{z}_1 - \sqrt{\hat{z}_2} + b\hat{z}_2 + e\sqrt{\hat{z}_1} \tag{6.15}$$

separating variables z_1 and z_2 allows to transform residual $r(t)$ into an equivalent system:

$$r(t) = G_a(\hat{z}_1) - G_b(\hat{z}_2) \tag{6.16}$$

where,

$$G_a(\hat{z}_1) = a\hat{z}_1 + e\sqrt{\hat{z}_1} \tag{6.17}$$
$$G_b(\hat{z}_2) = \sqrt{\hat{z}_2} - b\hat{z}_2 \tag{6.18}$$

therefore an optimal residual can be computed using $DTW_{on-line}$

$$\rho(t) = \mathbf{DTW_{on\text{-}line}}[G_a(\hat{z}_1), G_b(\hat{z}_2)] \tag{6.19}$$

Remark 1

$DTW_{on-line}$ can be applied just when an equivalent system, (6.13), can be found. But this is not always possible because sometimes the variables cannot be separated independently, as happens to the follow residual:

$$r(t) = a\hat{z}_1 - \sqrt{\hat{z}_2 - \hat{z}_1} + b\hat{z}_2 + e\sqrt{\hat{z}_1\hat{z}_2} \tag{6.20}$$

Remark 2

Note that missed detections can still occur if a wrong alignment of $G_a(\hat{z}_i)$ and $G_b(\hat{z}_i)$ when a fault is present could conceal its presence in the residual. Nevertheless, this not necessarily implies a higher missed detection rate. Since it only will occur when the variation in $\hat{z}_i$ provoked by a fault coincide with previous values of $\hat{z}_i$. Assuming an additive model for the fault trace in $\hat{z}_i$ described by $\hat{z}_i = \hat{z}_{io} + \hat{z}_{if}$, being $\hat{z}_{io}$ the normal expected value and $\hat{z}_{if}$ representing the fault, missed detection can appear in presence of short faults in specific instants, t, in which the following condition will be satisfied:

$$\hat{z}_{if}(t) = -\frac{d\hat{z}_{io}(t)}{dt}\frac{1}{\delta_i} \qquad \delta_i \leq \Delta \tag{6.21}$$

where, δ_i is the transmission delay and Δ is its maximum value which is assumed to be known. Thus, equation 6.21 can be used to provide information of minimum size of fault to be detected according to dynamics of measured variables and behavior of the network in terms of maximum delay. Moreover, missed detections are intrinsic to the selection of a neighborhood, $\mathcal{N}(0)$, instead of the equality in the evaluation of the residual.

6.5 Illustrative example: Fault detection in a NCS laboratory plant

6.5.1 Description of the system

The proposed approach has been used on a level control laboratory plant instrumented with a networked control and supervisory system. In this plant (See Figure 6.2) a pump sends liquid from one tank to another one that is placed in a higher level. Water returns to the first tank by the effect of gravity. A PID control maintains the level in the higher tank by governing the pump.

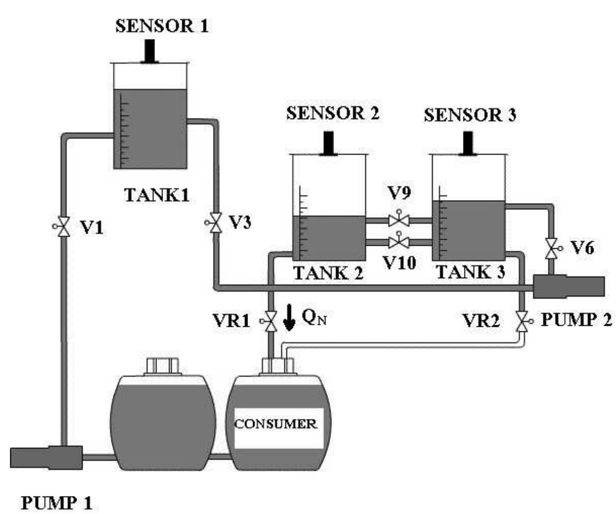


Figure 6.2: Laboratory plant

The system has been instrumented with sensors (flow-meter and level sensor connected to a SIEMENS ET200U module) and a PID controller (BÜRKET 1100) accessible both via PROFIBUS as independent nodes. A FDI (Fault Detection and Isolation) module has been implemented in a SCADA (Supervisory Control and Data Acquisition) system running in a computer accessing data from an OPC server connected to the PROFIBUS in a master/slave scheme as depicted in Figure 6.3. The time server update cycle (Time Rotation Time) is 137ms.

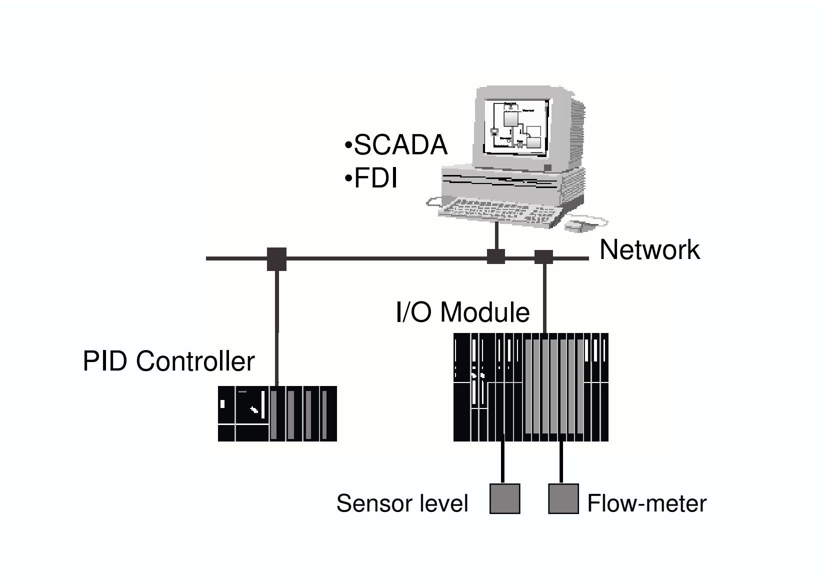


Figure 6.3: Process communication

The availability of data in the OPC-client computer has been tested under different conditions of refreshing time (Sample time) and number of devices.

Figure 6.4a) represents the histograms of delays of the monitored process variables with a sample time of 100ms. Delays were calculated as the difference between timestamps labelled by the OPC server. Samples located close to zero denotes repeated data, which means that the server did not have time to update the value. Samples around 100ms represent data arriving on time and samples on the right of the histogram indicate delayed data.

While for histogram of figure 6.4b), delays are insignificant compared to the sample time (1000ms). In that situation is guaranteed that the refreshing time in the OPC-client computer will coincide with OPC server updates.

It can be concluded that sample time and timestamp not are always coincident and obviously delays and misalignments increase as sample time decreases nearly to the target rotation time. In order to emphasize this source of errors residual computation has been evaluated under these circumstances.

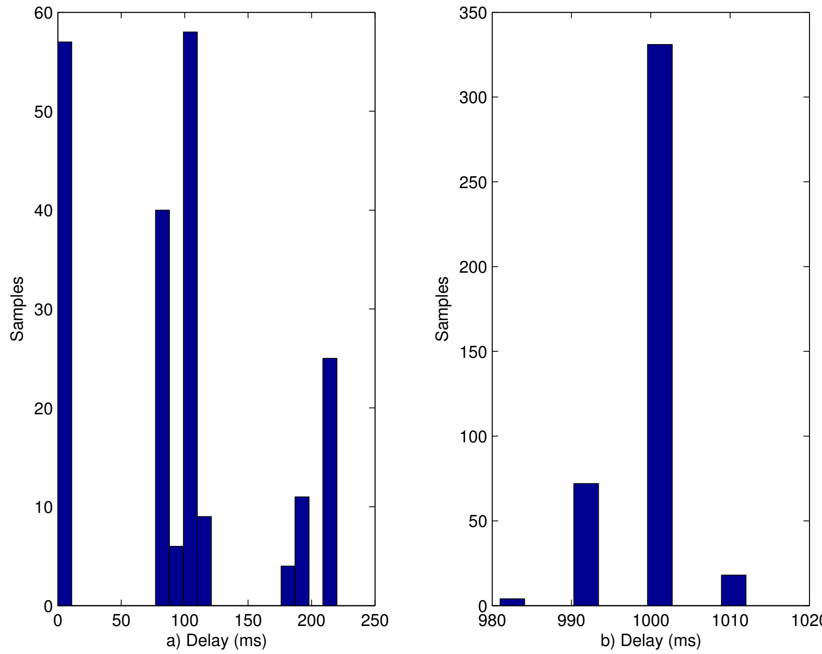


Figure 6.4: Histograms of delays of the monitored process variables a)Ts=100ms b)Ts=1000ms.

6.5.2 Results

Applying the structural analysis technique (Blanke et al., 2003) the following computational forms for the residuals (See details in Appendix C):

$$r_1 = Q_P - k_1\sqrt{h_3 - 13} - k_2\sqrt{h_3 - 7,5} - A\dot{h}_3 \tag{6.22}$$

$$r_2 = k_1\sqrt{h_3 - 13} + k_2\sqrt{h_3 - 7,5} - k_3\sqrt{h_2} - A\dot{h}_2 \tag{6.23}$$

$$r_3 = u(t).\bar{Q}_P - Q_P \tag{6.24}$$

where Q_P, h_2 and h_3 are the monitored process variables that represent the input flow into the tank 3, the level of thank 2 and the level of tank 3, respectively; all the monitored process variables are affected by the communication delays. A is the cross section area of the tanks and k_1, k_2 and k_3 are known parameters.

Residuals must be evaluated at anytime with values of process variables acquired at the same time. Typically a sampling time is defined to evaluate periodically the consistency of data coming from process. This periodicity also facilitates the computation of derivatives involved in the relations, i.e. r_2. Derivative calculation is done by subtracting the output value at the previous time step from the current value, and dividing by the sample time.

In order to see how misalignments affect residual r_2, a sample rate of 100ms has been fixed as a periodic interval to compute the redundancy equations. Decision thresholds have been determined as $\bar{x} \pm 3\sigma$, where $\bar{x}$ is the mean and σ is the standard deviation.

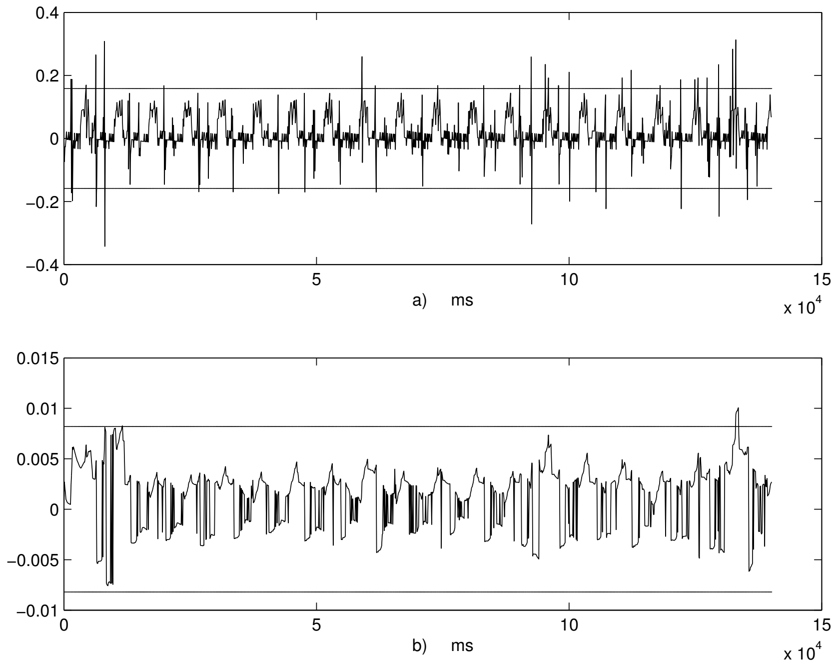


Figure 6.5: a) r_2 and b) ρ_2 using $DTW_{on-line}$.

Figure 6.5a) depicts the implementation of r_2 and it can be seen some values that cross the thresholds, which might lead to false alarms.

In order to apply $DTW_{on-line}$, r_2 has to separate in two signals:

$$G_a(h_3) = k_1\sqrt{h_3 - 13} + k_2\sqrt{h_3 - 7,5} \tag{6.25}$$
$$G_b(h_2) = k_3\sqrt{h_2} - A\dot{h}_2 \tag{6.26}$$

using $DTW_{on-line}$ the r_2 can be computed as follows:

$$\rho_2 = DTW_{on-line}(G_a(h_3), G_b(h_2)) \tag{6.27}$$

Figure 6.5b) shows how $DTW_{on-line}$ reduces the number of false alarms. The warping window size has been configured in 40 samples.

6.5.3 Time consuming

Table 6.8 resumes the time consuming comparison of two sequences using DTW and $DTW_{on-line}$. Algorithms implementation were done in a 2.4 GHz Pentium 4 with 512 MB of RAM. The computation of the cumulative distance matrix is the most consuming time factor of both methods, however since $DTW_{on-line}$ uses previously calculated it is faster than DTW.

Table 6.8: Time consuming comparison between DTW and $DTW_{on-line}$

Method	Warping window width	Cumulative distance time consuming	Path time consuming
DTW	40	1.1 s	1 ms
$DTW_{on-line}$	40	20 ms	1 ms

6.6 Conclusions

In this chapter, an approach based on classic DTW was developed to be used online in order to obtain residuals from a laboratory plant. This approach is specially suitable for those errors related with time distortions. Therefore, it will be useful for distributed systems with communication delays and for hybrid systems with on/off sensors or actuators causing misalignments between real and simulated signals.

The results show a hight robustness for on-line DTW. In fact, the results obtained evidenced less false alarms using on-line DTW than a normal implementation.

As restrictions, the new approach continues using dynamic programming and it could be computationally expensive depending on considerations as number of variables, warping window width, sample time or computer effort.

Chapter 7

Time Misalignment Reduction in Symptom Based Diagnosis

7.1 Introduction

In non model based fault diagnosis the deterministic model given by analytical equations are not available anymore. The only information we have from the process are the input $u(t)$ and the output $y(t)$ signals represented as a time-series which are a rich source information that can be used for diagnosis purpose.

A symptom is a subjective evidence of a fault that indicates the existence of a fault, then symptom based method allows a forward reasoning from fault to possible faults. This method can be performed with different techniques: fault trees, heuristics (rules), knowledge about process history (cases), statistical knowledge (belief networks), classification (neural network), expert systems, Case Based Reasoning.

Case Based Reasoning (CBR) is a problem solving methodology based on the reuse of previous experiences. The basic idea is focused on the hypothesis that "similar problems have similar solutions" (Wilke and Bergmann, 1998). CBR methodology proposes a four-step cycle (Retrieve, Reuse, Revise and Retain) also known as the 4R-cycle. It basically consists in Retaining experiences as cases for a further Reuse (submitted to a Revision procedure). Cases are registers containing a description of a problem "symptoms" and its solution "diagnostic". The aim is to reuse these cases for solving new problems by analogy. In presence of a new problem, the basic procedure consists of Retrieving analogue cases, according to their description (attributes defining symptoms), and reusing their solutions (diagnostic).

When cases are represented by time series, searching within a time series database for those series that are similar to a given query sequence is very important during the Retrieve step. Time misalignments may produce erroneous results when this task is carried out.

In this chapter Dynamic Time Warping (DTW) is proposed as a similarity criteria to implement the retrieval task, with the aim to reducing the influence of time-misalignments.

An electrical system problem, known as voltage sag, has been used to test the proposed method, (Llanos et al., 2003a; Meléndez et al., 2004a; Llanos et al., 2004b).

The chapter is organized as follows: Section 7.2 presents the methods of Case Based Reasoning (CBR). The efect of time misalignments while performing the Retrieving task is analysed in Section 7.3. Proposed solution using DTW is described in Section 7.5. The Illustrative example is shown in Section 7.6. Finally some concluding remarks are given in Section 7.7.

7.2 Case Based Reasoning

Case Based Reasoning is an approach to problem solving that is able to use specific knowledge of previous experiences (López and Plaza, 1997). A new problem is solved by matching it with a similar past situation. If the problem is solved, this new situation will be retained in order to solve other new ones. In the case of diagnosis, solving the problem means that the CBRsystem proposes a solution that is satisfactory enough to identify the new fault.

Many authors have discussed about, the appropriate situations where CBR offers advantages. In (Main et al., 2001) many of these advantages were summarized and the order in which they appear here is not indicative of their level of importance:

- Reduction of the knowledge acquisition task: by eliminating the extraction of model or set of rules as is necessary in model/based reasoning.

- To avoid repeating mistakes made in the past: system can use the information about what caused faults in the past to predict any faults in the future.

- Graceful degradation of performance: some model based systems cannot even attempt to solve a problem on the boundaries of its knowledge or scope, or when there is missing or incomplete data. In contrast case based systems can often have a reasonably successful attempt at solving these types of problem.

- *To be able to reason in domains that have not been fully understood, defined or modelled:* CBR can deal with only a set of cases from the domain.

- *To be able to make predictions:* the reasoner may be able to predict the success of the suggested solution to a current problem. This may be done by referring to the stored solutions and to the differences between the previous and current solution context.

- *To learn over time:* if cases are tested in the real world an level of success determined, there cases can be added into the case base to reason with it in future.

- *To reason in a domain with small body of knowledge:* a case based reasoner can start with a few known cases and incrementally increase its knowledge as cases are added to it.

- *To reason with incomplete or imprecise data and concepts:* as cases are retrieved not just when identical to the current query case but also when they are within some measure of similarity, incompleteness and imprecision can be deal with. While these factors may cause a slight degradation in performance due to the current and retrieved having increased disparity, reasoning can still continue.

- *Avoid repeating all the steps that need to be taken to arrive at a solution:* by reusing a previous solution, the steps taken to reach the retrieved solution can be reused themselves.

- *Provide a means of explanation:* by explaining how a previous case was successful in a situation, using the similarities between the cases and the reasoning involved in adaptation a CBR systems can explain its solution to a user

- *Can be used in different ways:* CBR can be used for many purposes as for creating a plan, making a diagnosis, arguing a point of view, etc.

- *Reflects human reasoning:* humans can understand a CBR system's reasoning and explanations and are able to be convinced of the validity of the solutions they are receiving.

7.3 CBR cycle

(Aamodt and Plaza, 1994) have described CBR as a cyclical process comprising the four REs. It basically consists in **R**etaining experiences as cases for a further **R**euse. Cases are registers containing a description of a problem and its solution. The aim is to reuse these cases for solving new problems by analogy. In presence of a new problem, the basic procedure consists of **R**etrieving analogue cases and reusing their solutions. Reuse implies an adaptation procedure of the retrieved solutions that is finished with a **R**evision. After validation, the cycle is completed by Retaining the solved situation (problem + new solution). In practice it is often confused the separation between reuse and revise and it is common to merge both in one operation named adaptation. The procedures involved in CBR can be represented by a schematic cycle, see Figure 7.1.

Cases are stored according to a suitable structure, named case memory, into case bases where indexing mechanisms are used for an efficient retrieval. CBR foundations and a detailed description of this methodology based on the 4R are wider described in (Aamodt and Plaza, 1994; Langseth et al., 1999). Next the four stages depicted in Figure 7.1 are described.

7.3.1 Retrieval

The Retrieval of cases is closely related and dependent on the indexing methods used. In general, several techniques are currently used: nearest-neighbor retrieval, inductive retrieval, knowledge guided approaches and validated retrieval.

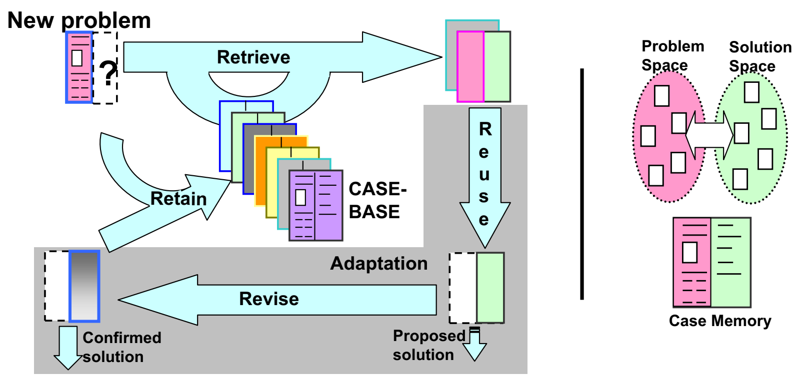


Figure 7.1: Cases and CBR cycle: Retrieve, Reuse, Retain and Revise.

- The most commonly used criteria for case retrieval are based on the concept of distance. They are used to obtain the **k-nearest neighbors** of a case, C_A, from a case base containing cases, designed by C_B, the indices A and B in the notation can be interpreted as **A**ctual case and any case from the case-**B**ase. The following is a general expression used for distance calculation between cases. They are supposed to be composed by a set of N attributes x_i which similarity is measured by a function, $sim()$:

$$sim(C_A, C_B) = \sum_{1}^{N} f(w_i) \cdot sim(x_i^A, x_i^A) \tag{7.1}$$

  Thus, a distance (or similarity) between two cases is obtained from the weighted distances between symptoms composing the two cases to be compared. Each symptom is weighted, w_i, according to its importance with respect to others in the case. The influence of w_i in the distance criteria can be applied with a multiplier effect or using a normalized function $f(w_i)$.

- An alternative retrieval technique used by many CBR tools involves a procedure called **induction**. Induction is a technique developed by machine learning researchers to extract rules or construct decision tree from past data. In CBR systems, the case-base is analyzed by an induction algorithm to produce a decision tree that classifies (or indexes) the cases. The most widely used induction algorithm in CBR tools is called ID3 (Quinlan, 1986).

- **Knowledge guided approaches**: knowledge guided approaches to retrieval use domain knowledge to determine the features of case which are important for that case in particular to be retrieved in future. In some situations different features of each case will have been important for the success level of that case.

- **Validated retrieval** (Simoudis, 1992), consists of two phases, firstly the retrieval of all cases that appear to be relevant to a problem, based on the main features of query case. The second phase involves deriving more discriminating features from the group of retrieved cases to determine whether they (cases) are valid in the current situation. The advantage of this method is that inexpensive methods can be used in the second phase as they are applied to only a subset of the case-base.

7.3.2 Reuse

Once a matching case is retrieved, a CBR system will attempt to **R**euse the solution suggested by the retrieved case. In many circumstances the solution may be sufficient. However, in other instances the solution from the retrieved case may be close to the required solution, but no close enough. The CBR system must then adapt the solution stored in the retrieved case to the needs of current case. Adaptation looks for prominent differences between the retrieved case and the current case and then applies rules that take those differences into account when suggesting a final solution. There are two principal methods to reuse (or adapt) past cases (Allemang, 1994):

- **Transformational reuse**: which consists of reusing past solutions. The retrieved case solution is not the solution adopted for the new problem, rather there is a transformation function that generates the new solution from past cases.

- **Derivational reuse**: this consists of reusing the procedure followed to deduce the past solutions. In this approach, the retrieved case must contain information about the method used for solving the retrieved problem, including a justification of the operations used, their subgoals, alternatives generated, failed search paths, etc. Derivational reuse reruns the retrieved method with the new case details and replays the old plan into the new context.

7.3.3 Revise

In this phase the proposed solution is analyzed to assess its accuracy. The proposed solution (generated in **R**euse) is evaluated, and if the solution is successful, the CBR can learn from it as will be shown in the next phase (case **R**etention); if the solution is not successful it can be repaired using domain specific knowledge. Evaluating a solution is normally done by applying the solution to a real world problem. Under some circumstances, the solution can be evaluated using a simulation. If the proposed solution needs to be repaired, it is necessary to identify the errors in the solution and to generate an explanation for them, using techniques that allows to repair errors.

7.3.4 Retain

The power of CBR is based on the richness of the case-base; therefore it is important to retain new cases and its solution. In this phase of the CBR cycle it is determined what can be learned and retained from a problem solving episode. The learning algorithm must take into account the outcome of the previous phase of the CBR cycle. The main sub-tasks of this step are:

- Selecting which information must be retained
- Establishing how this information must be saved
- Establishing how to index the new problem solving episode (case) for future retrievals
- Determining how to integrate the new case in the memory structure

7.4 Effects of time misalignment in case retrieval

7.4.1 Cases represented as time-series and experience

The conceptual definition of cases, considered in CBR, allows performing an association between two complementary views of process behaviour. One of them is provided by acquisition systems as flows of data that are systematically collected and stored. The other view is the human perception of process behaviour enhanced by the expertise and experience. They are two different views of the same reality (Langseth et al., 1999), the process, which must be combined in order to improve the global knowledge of process needed in assessment tasks. According to this premise, cases are conceived as knowledge containers that perform the association between both: $C = [S(t), D]$: where $S(t)$ represents symptoms that characterise the situations under study and D reflects the operator experience (diagnosis, hypothesis, etc). Symptoms are representations of acquired signals obtained directly from data or after an abstraction procedure. The expert view encapsulated in D represents the evaluation (diagnosis) of the given situation, a set of actions to perform under determined situation or similar information needed to preserve process under normal operation conditions or for warning operator in posterior similar situations.

In this Chapter symptoms are represented as time-series obtained directly from the system output $y(t)$ during a time window $[t_i, t_f[$. Therefore a case is defined as:

$$C = [y(t), D] \tag{7.2}$$

7.4.2 Time misalignment in case retrieval

As was introduced above in subsection 7.3.1, the most commonly used criteria for case retrieval are based on the concept of distance. Most of algorithm that operate with time-series of data use the Euclidean distance or some variation. However, Euclidean distance could produce an incorrect measure of similarity because it is very sensitive to distortions in the time axis (time misalignments). Therefore, robust yet sensitive methods for comparing unsynchronized signals are an active area of research.

7.5 Dynamic time warping for case retrieval

Since Dynamic Time Warping is a good method to compensate temporal distortions on the time axis, in this chapter it is proposed to use it as a distance criteria for case retrieval. Therefore applying **DTW** to (7.1), produces the similarity function:

$$sim(C_A, C_B) = \mathbf{DTW}(y_A(t), y_B(t)) \tag{7.3}$$

Thus, a distance (or similarity) between two cases (C_A, C_B) is obtained from DTW between theirs corresponding symptoms (y_A, y_B). Once a matching case is retrieved, a CBR system will attempt to **R**euse the solution (D) suggested by the retrieved case.

7.6 Application in an Electrical System: diagnosis of voltage sags in a 25kV substation

Electrical energy is supplied in form of a three-phase voltage of sinusoidal nature. There are four parameters that characterise the voltage wave, allowing measuring its degree of quality. These parameters are: frequency, amplitude, shape and symmetry. Electrical power stations produce a sine wave of 50 or 60 practically perfect cycles per second.

However, during the energy transport and distribution from the generation substation to the final customer, these parameters can suffer alterations which affect the wave quality. These alterations could have their origin in the electrical facility (as a result of breaker switching, failures, and so on), in natural phenomena, operation of loads (rectifying bridges, arc furnaces, and so on), or others. This alteration of the sinusoidal wave is usually transmitted to the electrical system (Bollen, 2000).

7.6.1 Voltage sag definition

An alteration defined in power systems is known as voltage sag. Standard definition (Bollen, 2000) of sags is based on the minimum rms value obtained during the event and its duration is the time interval between the instant when rms voltage crosses the voltage sag threshold (usually 90% of normal voltage) and the instant when it returns to normal level. A three-phase unbalanced voltage sag is shown in Fig. 7.2.

7.6.2 Cases represented by voltage magnitude and location

In order to establish a similarity criteria among the voltage sags, a technique proposed in (Bollen, 2000) to characterise them has been used. This technique allows a characterisation through a one complex voltage. The method is based on the decomposition of the voltage phasors in symmetrical components (Grainger and Stevenson, 1998). The three (complex) phase voltages in an unbalanced (not all three phases have the same magnitude or 120º between them) system can be completely described through three component voltages, known as symmetrical components. Positive-sequence voltage $\vec{V_1}$, negative-sequence voltage $\vec{V_2}$ and zero-sequence voltage $\vec{V_0}$ are calculated from the complex phase voltages as follows:

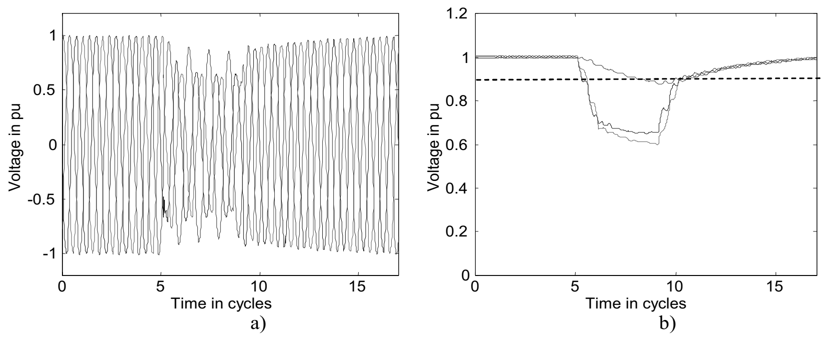


Figure 7.2: a)Example of a three-phase voltage sag b) rms voltage

$$\begin{pmatrix} \vec{V_0} \\ \vec{V_1} \\ \vec{V_2} \end{pmatrix} = \frac{1}{3} \begin{pmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{pmatrix}$$

where

$$a = -\frac{1}{2} + \frac{1}{2} j\sqrt{3} \tag{7.4}$$

The characteristic voltage, $\vec{V}$, indicates the severity of the sag, it is obtained from:

$$\vec{V} = \vec{V_1} - \vec{V_2}' \tag{7.5}$$

The voltage magnitude is a time-series defined as the absolute value of the characteristic voltage:

$$V = \mid \vec{V} \mid \tag{7.6}$$

Voltage sag magnitude, V, is used as symptom on the case representation. The evaluation (diagnosis) of the given situation is represented by the electrical location of the fault, L, namely **distribution** or **transmission**. Therefore a case may be represented by the double:

$$S = (V(t), L) \tag{7.7}$$

7.6.3 Dynamic Time Warping for voltage sag retrieval

Given two sags S_A and S_B described by their characteristic voltage V_A and V_B, respectively, the idea is to use DTW as a metric to define similarities between them:

$$sim(S_A, S_B) = DTW(V_A(t), V_B(t)) \tag{7.8}$$

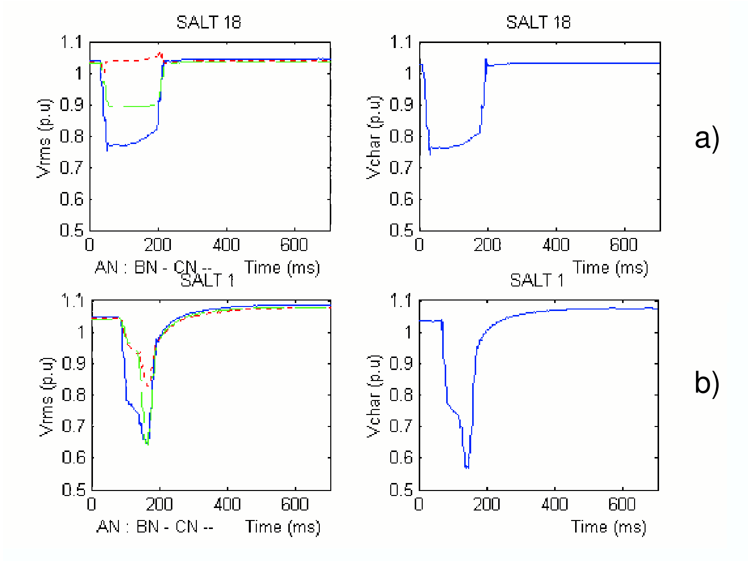


Figure 7.3: Voltage sag classification: a)Distribution b)Transmission.

7.6.4 Results

Spanish Electrical Facility (Endesa Distribution SL) has provided voltage sags in a 25kV distribution substation. 550 voltage sags were recorded and chosen to apply this method. 500 sags (of the recorded ones) were used for the creation of the case-base, they were separated in two groups according to the location of the fault (transmission/distribution): 360 voltages sags (72%) caused by fault in transmission systems and 140 voltage sags (28%) caused by faults in distribution systems. The 50 remaining voltages sags were used as new cases to be diagnosed.

Figure 7.3a) and 7.3b) show, respectively, the typical voltage sag wave forms according to the location of the fault, transmission or distribution. The rms voltage of the characteristic voltage sag is depicted on the right side of the figures. The rms voltage of the three phases are depicted on the left side of the figures. The rms voltage of the three phases were used in order to obtain the so called *temporal attributes*, (Llanos et al., 2003b; Mora et al., 2003a), temporal attributes are quantitative values such as magnitude, peaks, slopes and voltage sag duration. Temporal attributes are described in details in Appendix D.

In order to compare the results of DTW, two more distances criteria were also applied. One based on Euclidean distance, which also uses the time-series representation of the characteristic voltage, $V(t)$. And the other criteria was the Manhattan distance which is applied to the so called *temporal attributes*.

DTW is the less sensitive distance approach to the time misalignment. Table 7.1 shows the diagnostic error after applying the three above distance approaches to the 50 new cases.

Table 7.1: Diagnostic error using Euclidean, Manhattan and DTW distances

	Euclidean	Manhattan	DTW
Diagnostic error	22%	16%	8%

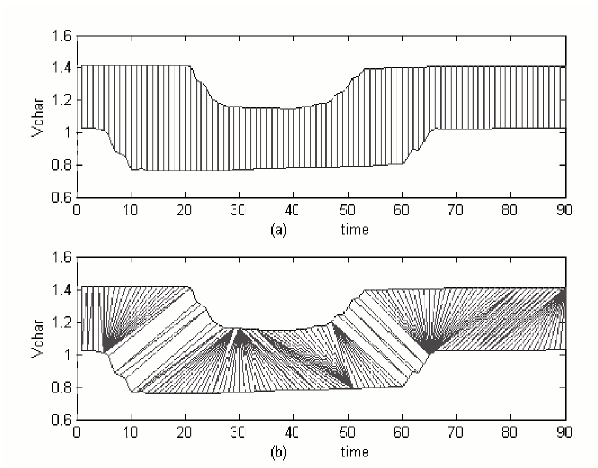


Figure 7.4: Two similar characteristic voltage sags. a) The Euclidean distance, b) DTW.

Figure 7.4a) shows the intuitive alignment produced by Euclidean distance. Both shapes have the same magnitude but they have been moved on the y-axes in order to show the alignment allowed by the two methods. The lines between the shapes show the point that is been compared in each waveform. Euclidean distance produces a pessimistic result of similarity since the signals are not aligned in time. While, Figure 7.4b) shows how DTW finds an alignment that allows a correct measure of similarity.

Figure 7.5 shows a new case that have not been well diagnosed using DTW. The waveform is more similar to a transmission voltage sag than a distribution one, but DTW gives a contrary result. It is because in this method magnitude is more relevant than voltage duration time.

The crucial observation is that the algorithm may try to explain variability in the Y-axis by warping the X-axis. This can lead to unintuitive alignments where a single point on one time series maps onto a large subsection of another time series.

The weakness of DTW is in the features it considers. It only considers a data points Y-axis value. For example if we consider two data points (x_i and y_j) which have identical values, but (x_i) is part of a rising trend and (y_j) is part of a falling trend. DTW considers a mapping between these two points ideal, although intuitively we would prefer not to map a rising trend to a falling trend.

To prevent this problem, in (Keogh and Pazzani, 2001), a modification of DTW that

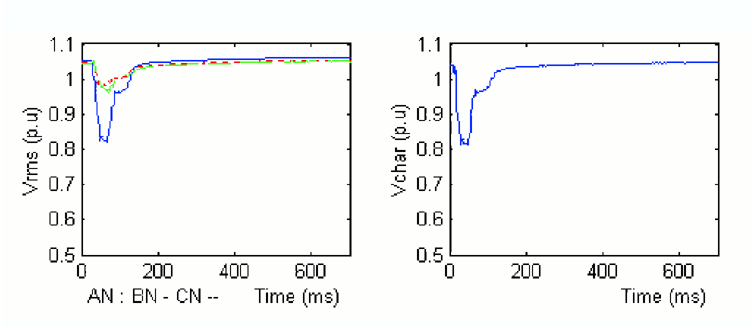


Figure 7.5: A wrong diagnosis of a new voltage sag.

does not consider the Y-values of the data points, but rather considers the higher level feature of "shape", was proposed. Information about shape by considering the first derivative of the sequences is obtained; this algorithm was called Derivative Dynamic Time Warping (DDTW).

As before we construct an n-by-m matrix where the (i^{th}, j^{th}) element of the matrix contains the distance $d(x_i, y_j)$ between the two points x_i and y_j. With DDTW the distance measure $d(x_i, y_j)$ is not Euclidean but rather the square of the difference of the estimated derivatives of xi and y_j. While there exist sophisticated methods for estimating derivatives, particularly if one knows something about the underlying model generating the data, we use the following estimate for simplicity and generality:

$$D_x[x] = \frac{(x_i - x_{i-1}) + ((x_{i+1} - x_{i11})/2)}{2} \tag{7.9}$$

This estimate is simply the average of the slope of the line through the point in question and its left neighbor, and the slope of the line through the left neighbor and the right neighbor. According to (Keogh and Pazzani, 2001), empirically this estimate is more robust to outliers than any estimate considering only two data points. Note the estimate is not defined for the first and last elements of the sequence. Instead we use the estimates of the second and penultimate elements respectively.

7.7 Conclusions

This chapter proposes the use of Dynamic Time Warping (DTW) for reducing the effects of time misalignment when Case Based Reasoning (a symptom based diagnosis) is performed. DTW is used as a similarity criteria to implement the retrieval task.

An electrical system problem, known as voltage sag, has been used to test the proposed method.

In order to compare the results of DTW, two more distances criteria were also applied. One based on Euclidean distance, which also uses the time-series representation of the characteristic voltage. And the other criteria was the Manhattan distance which has been applied to the temporal attributes. Results shows that DTW is the less sensitive distance approach to the time misalignment.

Chapter 8

Conclusions and Future Work

In the following paragraphs, the main aspects of the contributions of the present thesis are considered:

- Fault Detection and Diagnosis (FDD) deal with the timely detection, diagnosis and correction of abnormal conditions of fault in a process. To accomplish these tasks, all the process knowledge (structure, dynamic and static equations, parameters, heuristic knowledge, etc) and measured data from the process variables, have to be used. But these two sources of information carried several problems such as knowledge acquisition and representation, measurement noise, modelling uncertainties, time misalignment between measured data, among others. This thesis is focused on the study of time misalignments effects when performing FDD.

- As networks continue to built out, and network technology becomes cheaper and more reliable than traditional fixed point-to-point wiring connections, more and more control systems have operated and will operate over networks. In this case, the sensor, the actuator, diagnostic, command and coordination signals are all travelling over common networks. Feedback control systems with at least one loop closed through data networks are called *distributed control systems* (DCS). In distributed control systems, the residual computation form is implemented as an algorithm in one node of the network. In the literature review of FDI in DCS it is proposed the modification of the system model in order to deal with communication delays and data dropout. This thesis proposed a solution for time-delays and data dropout effects in DCS when performing FDI, the solution does not modify the system model. Our firsts studies of time misalignment are published in (Llanos et al., 2005; Llanos et al., 2004a).

- In this thesis it is shown that when transmission delays are known, it is possible to take them into account in the residual computation, thus introducing a delayed but otherwise unchanged decision and avoiding false alarms due to delays. However, when delays are unknown it is necessary to estimate them in order to compensate for their effect in the decision procedure. In this case, based on a very rough model of the delays, the problem is addressed as an optimization problem. A search algorithm is used for the delay estimation by minimizing the residual under the constraints given by the transmission model. When the persistent excitation condition is fulfilled, the delays estimation can be carried out giving reliable results and avoiding false alarms. The efficiency of the proposed approach has been applied on a control position for a DC motor. Results are published in (Llanos et al., 2007; Llanos et al., 2006).

- A simple formulation has been proposed to solve the problem of data dropout when performing FDI. It is based on the computation of the residual with the available data at each time instant instead of using the classical formulation. The principal benefit is the drastic reduction of false alarm ratio avoiding the classical trade off with missed detection. The efficiency of the proposed approach is illustrated on a laboratory plant, where the false alarms during the transient time window were reduced without increasing the missed detections under different conditions of data dropout.

- Another solution to the problem of time misalignment has been proposed. It is based on the Dynamic Rime Warping (DTW) algorithm. DTW is a technique that finds the optimal alignment between two signals. DTW is computationally expensive (in both time an memory) because it uses dynamic programming, therefore it is normally used for off-line applications. In this thesis we proposed a slight modification of DTW in order to adapt it for on-line application. The results show a hight robustness for on-line DTW. In fact, the results obtained evidenced less false alarms using on-line DTW than a normal implementation. Results are published in Gamero et al., 2004; Llanos et al., 2004b).

- The non model based techniques, that use the comparison and matching of signals for performing fault diagnosis, can also be affected by time misalignments because most of the algorithms, that operate with time-series of data, use the Euclidean distance or some variation. Euclidean distance could produce an incorrect measure of similarity because it is very sensitive to distortions in the time axis. This thesis proposed the use of DTW for reducing the effects of time misalignment when Case Based Reasoning (a symptom based diagnosis) is performed. DTW is used as a similarity criteria to implement the retrieval task. An electrical system problem, known as voltage sag, has been used to test the proposed method. In order to compare the results of DTW, two more distances criteria were also applied: One based on the Euclidean distance, which also uses the time-series representation and the Manhattan criterion. The results show that DTW is the least sensitive distance approach to the time misalignment. Results are published in (Meléndez et al., 2004a; Llanos et al., 2004b; Llanos et al., 2003a; Llanos et al., 2003b; Mora et al., 2003a; Mora et al., 2003c; Meléndez et al., 2003; Mora et al., 2003b).

Some research topics that can be studied in the future are discussed as follows:

- In order to consider the dynamic of the residual generator, any observer based residual generator could be used when analyzing the impact of network delay in residual generation.

- In this thesis the network communication model was modelled by a very simple deterministic model. It will be interesting to model phenomena as network queues, varying network loads, measurement noise and others peculiarities of networks that could degrade the performance of the fault diagnosis system.

- In this thesis the effects of time misalignments when performing fault isolation has not been analyzed. It would be interesting to analyze what happens to the signature matrix and even if its possible to design structured residuals sensitives to the network communication problems.

- The on line DTW approach continues using dynamic programming and it could be computationally expensive depending on considerations such as the number of variables, warping window width, sample time or computer effort. Future work can be done in order in to introduce restrictions that would allow reducing the effects of the above mentioned considerations.

- Another weakness of DTW is in the features it considers. DTW only takes into account a data points Y-axis value. In order to deal with this situation, in (Keogh and Pazzani, 2001), a modification of DTW that does not consider the Y-values of the data points, but rather considers the higher level feature of "shape", was proposed. This algorithm was called Derivative Dynamic Time Warping (DDTW). The next step should be to adapt DDTW in order be used on-line.

Appendix A

Design of the analytical redundancy relation of a control position for a DC motor

From Figure A.1 the following equations can be written based on Newton's law combined with Kirchhoff's law:

$$R_a i_a(t) + K_3 \frac{dc(t)}{dt} = K_0 K_1 e(t) \tag{A.1}$$

$$D \frac{d^2 c(t)}{d^2 t} + b_0 \frac{dc(t)}{dt} = K_2 i_a(t) \tag{A.2}$$

where, $K_0, K_1, K_2, K_3, b_0, D$ and R_a are known parameters and, $e(t)$ and $c(t)$, are measurements available from the process.

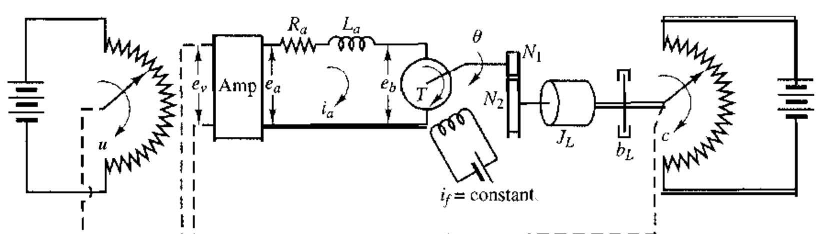


Figure A.1: Schematic diagram of a control position for a DC motor.

Table A.1: Incidence matrix.

$\nearrow$	i_a	e	c	$\dot{c}$	$\ddot{c}$
c_1	1	1		1	
c_2	1			1	
d_3				1	
d_4					1
m_1			1		
m_2		1			

Next equations describe the operation mode of the system:

$$c_1 \quad : \quad e(t) = \frac{R_a i_a(t) + K_3 \frac{dc(t)}{dt}}{K_0 K_1} \tag{A.3}$$

$$c_2 \quad : \quad i_a(t) = \frac{D \frac{d^2 c(t)}{d^2 t} + b_0 \frac{dc(t)}{dt}}{K_2} \tag{A.4}$$

$$d_3 \quad : \quad \dot{c}(t) = \frac{dc(t)}{dt} \tag{A.5}$$

$$d_4 \quad : \quad \ddot{c}(t) = \frac{d^2 c(t)}{d^2 t} \tag{A.6}$$

$$m_1 \quad : \quad c(t) = c_{m_1} \tag{A.7}$$

$$m_2 \quad : \quad e(t) = e_{m_2} \tag{A.8}$$

Redundancy relations have been obtained from the following matching Table A.1 using the ranking algorithm described in (Blanke et al., 2003). The corresponding oriented graph is shown in figure C.1.

Simplifying the operation model equations using the matched variables results in the

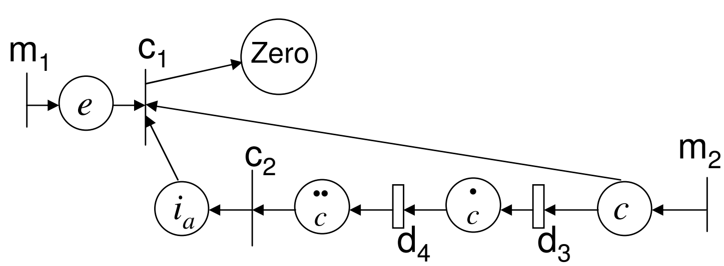


Figure A.2: Oriented structure graph of the system following redundancy relation:

$$r(t) = D\frac{d^2c(t)}{dt^2} + B\frac{dc(t)}{dt} - Ke(t) \tag{A.9}$$

whit,

$$B = b_0 + \frac{K_2K_3}{R_a} \tag{A.10}$$

$$K = K_0K_1K_2 \tag{A.11}$$

Appendix B

Design of the analytical redundancy relations of the laboratory plant

Next equations describe the model of the laboratory plant, see Figure B.2:

$$c_1 \quad : \quad Q_i = u.\bar{Q}_i \tag{B.1}$$

$$c_2 \quad : \quad Q_o = K.\sqrt{h} \tag{B.2}$$

$$d_3 \quad : \quad \dot{h} = \frac{d}{dt}h \tag{B.3}$$

$$c_4 \quad : \quad \dot{h} = \frac{1}{A}(Q_i - Q_o) \tag{B.4}$$

$$m_1 \quad : \quad h = h_{m_1} \tag{B.5}$$

$$m_2 \quad : \quad Q_i = Q_{i,m_2} \tag{B.6}$$

where h, Q_i and Q_o are the level, the input and the output flow to the higher tank, respectively. u is the control signal and $\bar{Q}_i$ is a parametrization of the pump. A and K are known parameters. m_1 and m_2 are additional measurement constraints given by the flow-meter and the level sensor. Redundancy relations have been obtained from the following matching table (Table B.1) where the '1' indicates the presence of the variable (columns) in the equation (rows). The ranking algorithm described in (Blanke et al., 2003) has been used to obtain the 'zero ' or redundancy equations. Circles indicate the selected variables in the iterative ranking procedure. The corresponding oriented graph (Figure B.1) shows how equation c1 and c4 are the basis for the two ARR.

Simplifying the operation model equations using the matched variables results in the following residuals:

$$r_1 = Q_i - K\sqrt{h} - A\dot{h} = 0 \tag{B.7}$$

$$r_2 = u.\bar{Q}_i - Q_i = 0 \tag{B.8}$$

After a simple discretisation (unitary sample time) of variables and derivatives the computation equation of residuals are the following:

$$r_1(k) = Q_i(k) - K\sqrt{h(k)} - A(h(k) - h(k-1)) = 0 \tag{B.9}$$

Table B.1: Incidence matrix.

$\nearrow$	Q_i	Q_o	h	$\dot{h}$
c_1	1			
c_2		1	1	
d_3			1	1
c_4	1	1		1
m_1			1	
m_2	1			

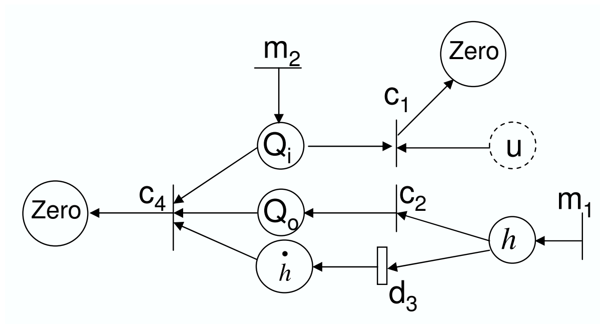


Figure B.1: Oriented structure graph of the system.

$$r_2(k) = u(k).\bar{Q}_i - Q_i(k) = 0 \tag{B.10}$$

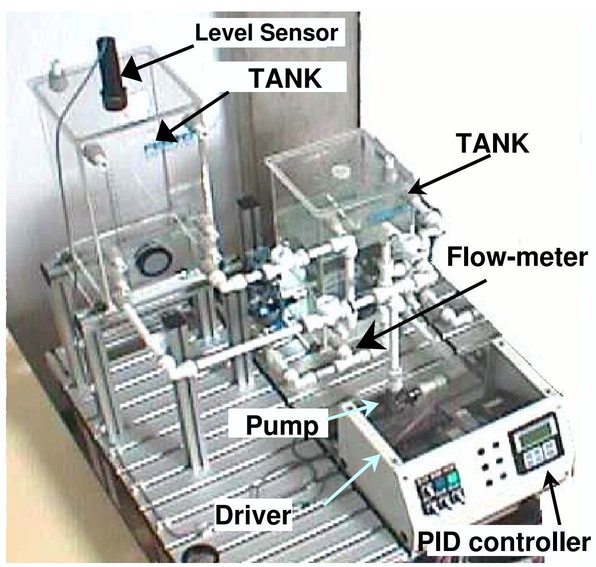


Figure B.2: Image of the laboratory plant.

Appendix C

Design of the analytical redundancy relations of the three tanks system

Next equations describe the model of the laboratory plant, see Figure C.2. Valves V9 and V10 are located $13cm$ and $7.5cm$, respectively, from the bottom of thanks. Depending on the water levels there exist several different operation modes for the system. As a practical example, level in TANK 3 has been maintained in $20cm$ and TANK 2 in $7cm$. Next equations describe the operation mode for the system:

$$c_1 \;:\; Q_L = 0 \tag{C.1}$$
$$c_2 \;:\; Q_P = u(t).\bar{Q}_P \tag{C.2}$$
$$c_3 \;:\; \dot{h}_3 = \frac{1}{A}(Q_P - Q_L - Q_{32}) \tag{C.3}$$
$$d_4 \;:\; \dot{h}_3 = \frac{d}{dt}h_3 \tag{C.4}$$
$$c_5 \;:\; Q_{32} = k_1\sqrt{|h_3 - 13|} + k_2\sqrt{|h_3 - 7,5|} \tag{C.5}$$
$$d_6 \;:\; \dot{h}_2 = \frac{d}{dt}h_2 \tag{C.6}$$
$$c_7 \;:\; \dot{h}_2 = \frac{1}{A}(Q_{32} - Q_N) \tag{C.7}$$
$$c_8 \;:\; Q_N = k_3\sqrt{h_2} \tag{C.8}$$
$$m_1 \;:\; h_3 = h_{3,m_1} \tag{C.9}$$
$$m_2 \;:\; h_2 = h_{2,m_2} \tag{C.10}$$
$$m_3 \;:\; Q_P = Q_{P,m_3} \tag{C.11}$$

where Q_{32} is the flow between tanks, Q_N is the output flow of the TANK 2, Q_p is the input flow into TANK 3 and Q_L appears when a leakage in TANK 3 occurs. A, k_1, k_2 and k_3 are known parameters. m_1, m_2 and m_3 are additional measurement constraints.

Redundancy relations have been obtained from the following matching table (table C.1) using the ranking algorithm described in (Blanke et al., 2003). The corresponding oriented graph is shown in figure C.1.

Simplifying the operation model equations using the matched variables results in the following redundancy relations:

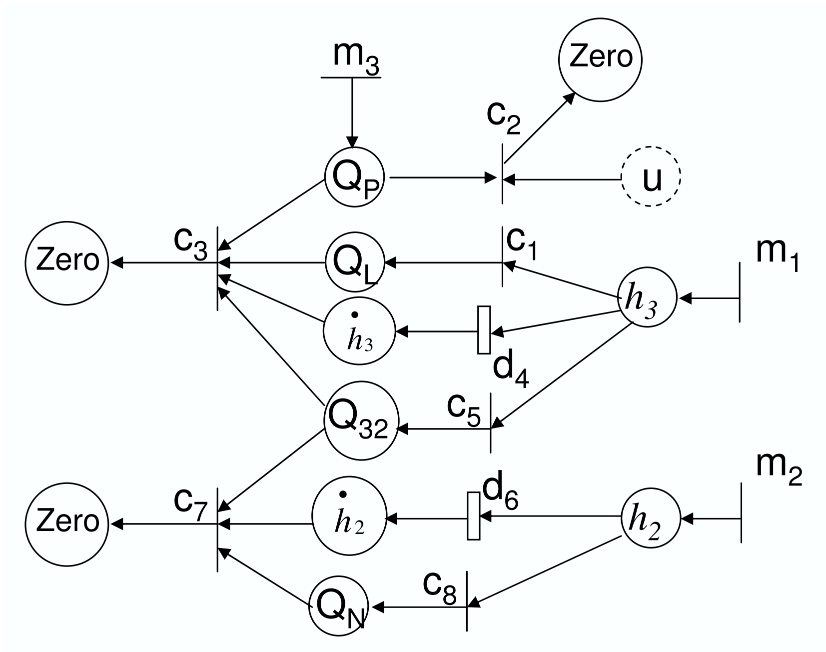


Figure C.1: Oriented structure graph of the three tanks system.

Table C.1: Incidence matrix.

$\nearrow$	Q_L	Q_P	$\dot{h}_3$	h_3	Q_{32}	$\dot{h}_2$	h_2	Q_N
c_1	1							
c_2		1						
c_3	1	1	1		1			
d_4			1	1				
c_5				1	1			
d_6						1	1	
c_7					1	1		1
c_8							1	1
m_1				1				
m_2							1	
m_3		1						

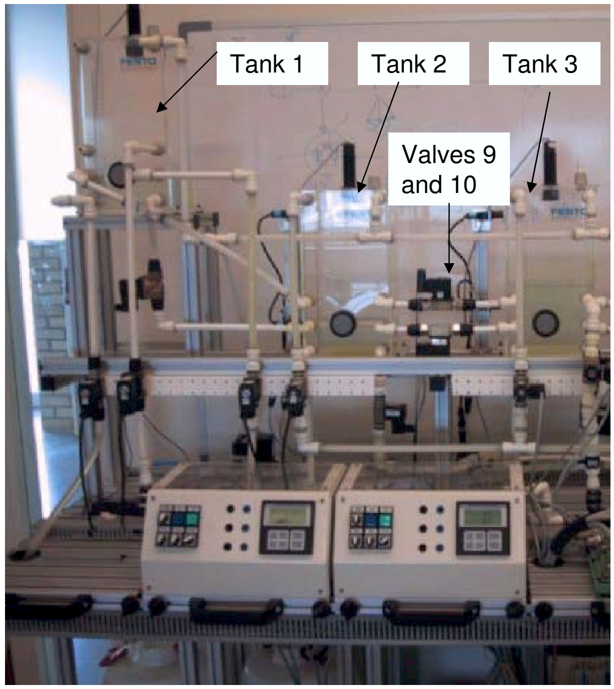


Figure C.2: Image of the laboratory plant.

$$
\begin{aligned}
Q_P - Q_{32} - A\dot{h}_3 &= 0 &\text{(C.12)}\\
Q_{32} - Q_N - A\dot{h}_2 &= 0 &\text{(C.13)}\\
u(t).\bar{Q}_P - Q_P &= 0 &\text{(C.14)}
\end{aligned}
$$

replacing equations (C.5), (C.8) and (C.11) in equations (C.12), (C.13) and (C.14), the analytical redundancy relations are expressed as follows:

ARR 1:

$$Q_{P,m_3} - k_1\sqrt{h_3 - 13} - k_2\sqrt{h_3 - 7,5} - A\dot{h}_3 = 0 \tag{C.15}$$

ARR 2:

$$k_1\sqrt{h_3 - 13} + k_2\sqrt{h_3 - 7,5} - k_3\sqrt{h_2} - A\dot{h}_2 = 0 \tag{C.16}$$

ARR 3:

$$u(t).\bar{Q}_P - Q_{P,m_3} = 0 \tag{C.17}$$

Appendix D

Temporal attributes of the voltage sags

Voltage sags have been characterised by using temporal attributes. Temporal attributes are obtained from measurements of the duration and magnitude of the voltage sags. The attributes have been divided in three-phase and single-phase temporal attributes and they are described as follows:

Three-phase Temporal Attributes

Three-phase temporal attributes are depicted in Fig. D.1.

Three-phase sag magnitude: Defined as the maximum reduction of voltage of three-phase power system during the sag.

Three-phase sag duration: Defined, as the maximum time during the rms voltage in a three-phase power system, is lower to 0.9 p.u.

Single-phase Temporal Attributes

Single-phase temporal attributes are depicted in Fig. D.2.

Single-phase sag magnitude (h): Maximum voltage reduction in every single phase of the power system.

Single-phase sag duration: Maximum time during the rms voltage is lower to 0.9 p.u. in every single phase of the power system.

Voltage fall slope: Slope at the beginning of the sag until a steady state is reached within a 2% band.

Voltage recovery slope: Slope in the end of the sag until a steady state is reached within a 2% band.

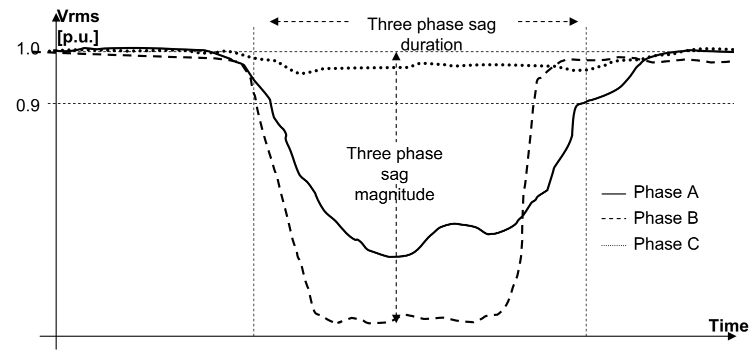


Figure D.1: Three-phase voltage sags attributes.

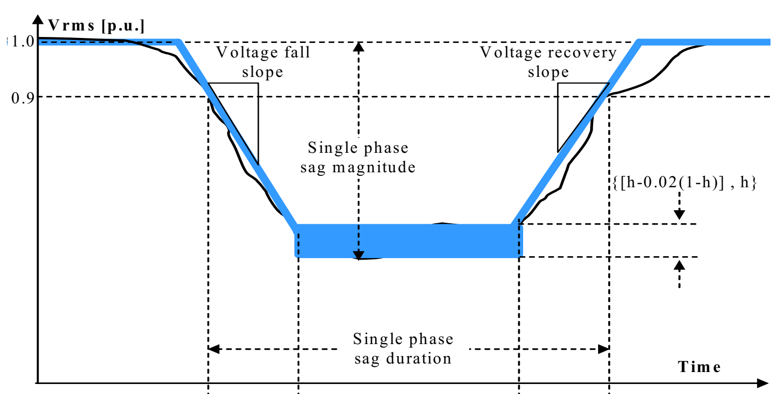


Figure D.2: Single-phase voltage sags attributes.

Manhattan distance for voltage sags comparison

Manhattan distance has been used to compare similarities between two sags S_A and S_B, where sag S, is defined by the vector of n attributes (X_i, with i=1..n). Thus, the distance between two sags S_A and S_B) can be computed as a weighted addition of local distances, defined by equation(D.1), between normalised attributes.

$$DIST(S_A, S_B) = \sum_{i=1} W_i dist(\overline{X_i^A}, \overline{X_i^B}) \tag{D.1}$$

with,

$$dist(\overline{X_i^A}, \overline{X_i^B}) = |\overline{X_i^A} - \overline{X_i^B}| \tag{D.2}$$

Normalisation is performed according to the range of each attribute (X_i min - X_i max). The following expression has been used with this purpose:

$$0 < \frac{X_i - X_i min}{X_i max - X_i min} < 1 \tag{D.3}$$

Voltage sag magnitude and voltage sag duration have been normalised according to the categories and typical characteristics of power system electromagnetic phenomena(IEEE Std 1159-1995), where voltages sag (included in short duration variations category) is defined with a typical duration of (0.5-30 cycles) and typical voltage magnitude of (0.1-0.9 p.u.).

Table D.1: Reference values for temporal attributes normalisation.

Three-phase temporal attributes		Single-phase temporal attributes			
Magnitude	Duration	Magnitude	Duration	Fall slope	Recovery slope
0.1-0.9 pu	0.5-30 cycles	*	*	*	*

* : those values have been selected from the maxim and minimum of real measures. And W_i has been used to weight the importance of each attribute in the global expression.

Figure D.3 shows an application developed with Microsoft Office Access. This application allows to record information as temporal and phasorial attributes, voltage and current r.m.s. waveforms, and information relates to the voltage sag location and system configuration. This application was used for the automatic management of voltage sags (Meléndez et al., 2004b) and also for classification of sags using fussy tool, for more details refers to (Mora et al., 2003c; Meléndez et al., 2003; Mora et al., 2003b).

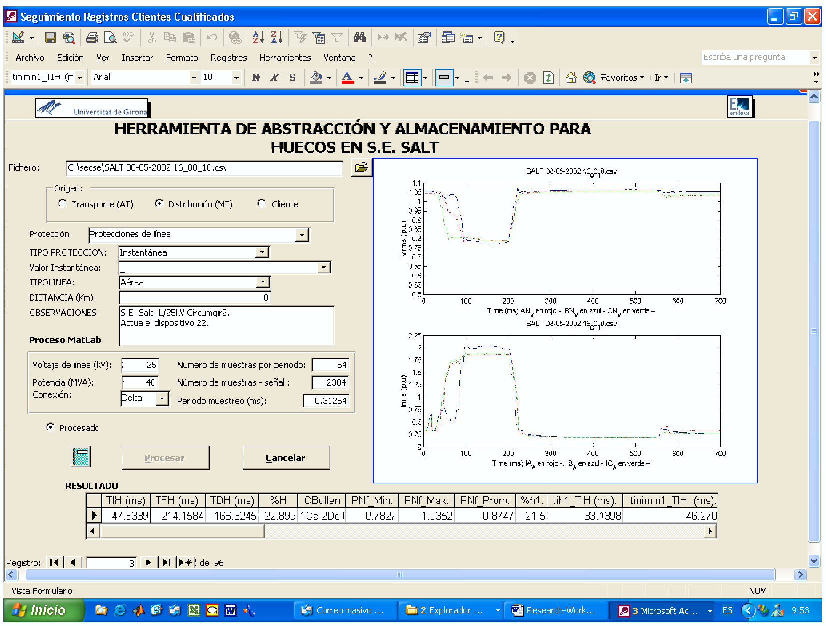


Figure D.3: Developed Tool for voltage sags registration

Bibliography

Aamodt, A. and Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications, IOS Press*, 1:39–59.

Allemang, D. (1994). Combining case-based reasoning and task-specific architectures. *IEEE Expert: Intelligent Systems and Their Applications*, 9(5):24–34.

Armengol, J. (1999). *Applications of the modal interval analysis to the simulation of the behaviour of dynamic systems with uncertain parameters*. PhD thesis, University of Girona.

Basseville, M. (1988). Detecting changes in signals and systems - a survey. *Automatica*, 24(3):309–326.

Basseville, M. and Nikiforov, I. (1993). *Detection of Abrupt Changes : Theory and Application*. Prentice Hall, New York.

Beard, R. V. (1971). Failure accommodation in linear systems through reorganization. Man Vehicle Lab Rep. MVT-7-1, Mass.Inst. Techno.

Berndt, J. and Clifford, D. (1994). Using dynamic time warping to find patterns in time series. pages 229–248. AAAI-94 Workshop on Knowledge Discovery in Database.

Blanke, M. (2005). Fault tolerant sensor fusion with an application to ship navigation. Prgaue, Czech Republic. 16th International Federation of Automatic Control World Congress IFAC'05.

Blanke, M., Kinnaert, M., Lunze, J., and M.Staroswiecki (2003). *Diagnosis and Fault-Tolerant Control*. Springer Verlag, Germany.

Bollen, M. H. (2000). *Understanding Power Quality problems*. IEEE Press series on power engineering, New York.

Bonarini, A. and Bontempi, G. (1994). A qualitative simulation approach for fuzzy dynamical models. *ACM Transactions on Modeling and Computer Simulation*, 4(4):258–313.

Brodsky, B. E. and Darkhovsky, B. S. (2000). *Non-parametric Statistical Diagnosis. Problems and Methods*. Kluwer Academic Publishers, ISBN 0-7923-8411-3.

Calado, J. and Sà da Costa, J. (1999). On line fault detection and diagnosis based on a coupled system. European Control Conference.

Cassar, J. and Staroswiecki, M. (1997). A structural approach for the design or failure detection and identification systems. pages 329–334, Belfort, France. IFAC/IFIP/IMACS Conference on Control of Industrial Processes.

Chen, J. and Liao, C. M. (2002). Dynamic process fault monitoring based on neural network and PCA. *Journal of Process Control*, 12(2):277–289.

Chen, J. and Patton, R. J. (1999). *Robust Model-based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers.

Chiang, L. H. and Braatz, R. D. (2003). Process monitoring using causal MAP and multivariate statistics: fault detection and identification. *Chemometrics and Intelligent Laboratory Systems*, (65):159–178.

Chittaro, L., Guida, G., Tasso, C., and Toppano, E. (1993). Functional and teleological knowledge in the multimodeling approach for reasoning. *IEEE Transactions on systems, man and cybernetics*, 23(6):1718–1751.

Chouaib, B. (2004). *Data Reconciliation as a Framework for Chemical Processes Optimization and Control*. PhD thesis, Department of Chemical Engineering, Polytechnical University of Catalunya, Spain.

Chow, E. Y. and Willsky, A. S. (1984). Analytical redundancy and the design of robust failure detection systems. *IEEE Transactions on Automatic Control*, AC-29:603–614.

Colomer, J. (1998). *Representació qualitativa asíncrona de senyals per a la supervisió de sistemes dinàmics*. PhD thesis, University of Girona, Thesis dissertation.

Colomer, J., Meléndez, J., and Gamero, F. I. (2002). Pattern recognition based on episodes and dtw. application to diagnosis of a level control system. Catalunya, Spain. 6th International Workshop on Qualitative Reasoning.

Control-Eengineering-Practice (1997). Special volume on supervision, fault detection, and diagnosis of thecnical systems. *Control Engineering Practice*, 5(5).

Cordier, M., Dague, P., Lévy, F., Montmain, J., Staroswiecki, M., and Travé-Massuyè, L. (2004). Conflicts vesus analytical redundancy relations: A comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives. *IEEE Trans. on Syst, Man and Cyber- Part B:Cybernetics*, 34(5):2163–2177.

Dague, P. (1995). Qualitative reasoning: a survey of techniques and applications. *AICOM*, (8).

De Kleer, J., Mackworth, A., and Reiter, R. (1992). Characterizing diagnoses and systems. *Artif. Intell.*, 56(2-3):197–222.

De Kleer, J. and Williams, B. (1987). Diagnosing multiple faults. *Artificial Intelligence*, 32(1):97–130.

De la Fuente, M. and Represa, C. (1997). A comparative study of neural networks based approach for fault detection. IFAC Safeprocess 1997.

Delmaire, G., Cassar, J. P., and Staroswiecki, M. (1994a). Comparison of identification and parity space approaches for failure detection in single input single output systems. *Proceedings of the Third IEEE Conference on Control Applications*, 2:865 – 870.

Delmaire, G., Cassar, J. P., and Staroswiecki, M. (1994b). Identification and parity space techniques for failure detection in siso systems including modelling errors. *Proceedings of the 33rd IEEE Conference on Decision and Control*, 3:2279 – 228.

Ding, S. X., Ding, E. L., and Jeinsch, T. (2002). A new optimization approach to the design of fault detection filters. In *Proc. IFAC SAFEPROCESS 2000*, pages 250–255, Budapest, Hungary,. SAFEPROCESS2000.

Ding, S. X., Jeinsch, T., Frank, P., and Ding, E. L. (2000). A unified approach to the optimization of fault detection systems. *Int. J. Adaptive Contr. Signal Processing*, 14(7):725–745.

Ding, S. X. and Zhang, P. (2005). Observer-based monitoring of distributed networked control systems. Ajaccion, France. 1st Workshop on Networked Control Systems and Fault Tolerant Control.

Ding, X. and Jeinsch, T. (1999). An approach to analysis and design of observer and parity relation based fdi systems. 14th IFAC World Congress.

Dong, Y., Qing-Long, H., and Chen, P. (2004). State feedback controller design of networked control systems. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 51, Issue:640–644.

El-Farra, N. H., Gani, A., and D.Christofides., P. (2005). Fault-tolerant control of process systems using communication networks. *AIChE Journal*, 51, No. 6:1665–1682.

Fang, H., Ye, H., and Zhong, M. (2006). Fault diagnosis of networked control systems. China. IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes.

Fang, H., Ye, H., and Zhong, M. (2007). Fault diagnosis of networked control systems. *Annual Reviews in Control*, 31:55–68.

Frank, P. and Ding, X. (1997). Survey of robust residaul generation and evaluation methods in observer-based fault detection systems. *Journal of Process Control*, 7(6):403–424.

Frank, P. M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey and some results. *Automatica*, 26:459–474.

Frank, P. M. (1996). Analytical and qualitative model-based fault diagnosis - a survey and some new results. *European Journal of Control*, 2:6–28.

Gamero, F. I., Llanos, D., Colomer, J., and Meléndez, J. (2004). Residual computation and evaluation using dynamic time warping. Genoa, Italy. IMACA.

Gertler, J. (1995). Diagnosing parametric faults-from identification to parity relations. pages 1615–1620, Seattle, WA. American Control Conference.

Gertler, J. (1998). *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, New York.

Gertler, J. (2000). All linear mehtods are equal and extendible to nonlinearities. pages 52–63, Budapest, Hungary. IFAC Sefeprocess.

Gertler, J., Costin, M., Fang, X., Kowalczuk, Z., Kunwer, M., and Monajemy, R. (1995). Model based diagnosis for automotive engines-algorithm development and testing on a production vehicle. *IEEE Transactions on Control Systems Technologies*, 3:61–69.

Gertler, J., Fang, X., and Luo, Q. (1990). Detection and diagnosis of plant failures: the orthogonal parity equation approach. *Control and Dynamic Systems*, 37:159–216.

Gertler, J. and Singer, D. (1990). A new structural framework for parity equation-based faiulure detection and isolation. *Automatica*, 26:381–388.

Gertler, J. J. (1991). Analytical redundancy methods in fault detection and isolation. pages 9–21. IFAC Symposium Safeprocess.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, USA.

Grainger, J. J. and Stevenson, W. D. (1998). *Análisis de Sistemas de Potencia*. Mexic.

Grinstead, C. M. and Snell, J. L. (1997). *Introduction to Probability*. 2nd edition.

Hamelin, F., Sauter, D., and Aubrun, M. (1994). Fault diagnosis in systems using directional residuals. pages 3040– 3045. 33rd IEEE Conference on Decision and Control.

Hamscher, W., Console, L., and de Kleer, J. (1992). Readings in model based diagnosis.

Hu, S. and Zhu, Q. (2003). Stochastic optimal control and analysis of stability of networked control systems with long delay. *Automatica*, (39):1877–1884.

Iserman, R. (1997). Supervision, fault detection and fault-diagnosis methods-an introduction,. *Control Engineering Practice*, 5(5):639–652.

Isermann, R. (1984). Process fault detection based on modeling and estimation methods. *Automatica*, 20:387–404.

Isermann, R. and Ballé, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical process. *Control Engineering Practice*, 5(5):709–719.

Jalote, P. (1994). *Fault Detection and Diagnosis in Engineering Systems*. Prentice Hall, ISBN: 0133013677.

Jiang, B., Staroswiecki, M., and Cocquempot, V. (2002a). Fault identification for a class of timedelay systems. In *Proc. Amer. Contr. Conf.*, pages 8–10, Anchorage, USA,. Amer. Contr. Conf.

Jiang, B., Staroswiecki, M., and Cocquempot, V. (2002b). h_∞ fault detection filter for a class of discrete-time systems with multiple time delays. In *Proc. 15th IFAC World Congress*, Barcelona, Spain. 15th IFAC World Congress.

Jiang, B., Staroswiecki, M., and Cocquempot, V. (2003). h_∞ fault detection filter design for linear discrete-time systems with multiple time delays. *Int. J. of Systems Science*, 34(5):365–373.

Jiang, C. and Zhou, D. H. (2005). Fault detection and identification for uncertain linear time-delay systems. *Computers & Chemical Engineering*, (30):228–242.

Johannessen, S. (2004). Time synchronization in a local area network. *IEEE Control Systems Magazine*, 24, Issue:61–69.

Kano, M., Hasebe, S., Hashimoto, I., and Ohno, H. (2001). A new multivariate statistical process monitoring method using principal component analysis. *Computers and Chemical Engineering*, 27(7-8):1103–1113.

Kassidas, A., MacGregor, J. F., and Taylor, P. A. (1998a). Synchronisation of batch trajectories uring dynamic time warping. *AIChEJ*, 44(4):864–875.

Kassidas, A., Taylor, P. A., and MacGregor, J. F. (1998b). Off-line diagnosis of deterministic faults in continuous dynamic multivariable processes using speech recognition methods. *J. Process Control*, 8(5-6):381–393.

Keogh, E. and Pazzani, M. (2001). Derivative dynamic time warping. Chicago, USA. In Proc. of the First Intl. SIAM Intl. Conf. on Data Mining.

Krzanowski, W. J. (1979). Between-group comparison of principle components. *J. Am. Stat. Assoc.*, 74(367):703–707.

Kuipers, B. (1994). Qualitative reasoning-modelling and simulation with incomplete knowledge. *MIT Press*.

Kuipers, B. (2001). Qualitative simulation. In Robert A. Meyers, E.-i.-C., editor, *Encyclopedia of Physical Science and Technology, Third Edition*, pages 287–300.

Kuipers, B. J. (1993). Reasoning with qualitative models. *Artificial Intelligence*, 59:125–132.

Langseth, H., A., A., and M, W. O. (1999). Learning retrieval knowledge from data. In Anand, A. and Aha, editors, *IJCAI'99 Workshop ML-5: Automating the construction of Case-Based Reasoners*, pages 77–82.

Lee, D., Allan, J., Thompson, H. A., and Bennett, S. (2001). Pid control for distributed system with a smart actuator. *Control Engineering Practice*, 9 (11):1235–1244.

Li, Y., Wen, C. L., Xie, Z., and Xu, X. H. (2004). Synchronisation of batch trajectory based on multi-scale dynamic time warping. pages 2403–2408. 2th International Conference on Machine Learning and Cybernetics (ICMLC 2003).

Li, Z. and Fang, H. (2006). A novel controller design and evaluation for networked control systems with time-variant delays. *Journal of the Franklin Institute*, 342(2):164–167.

Lian, F., Moyne, J., and Tilbury, D. (2003). Modelling and optimal controller design of networked control systems with time delays. *International Journal of Control*, 76(6):591–606.

Llanos, D., Colomer, J., Meléndez, J., and Staroswiecki, M. (2005). Influence of network delays in residual computation. Prague, Czech Republic. 16th IFAC World Congress.

Llanos, D., Cuberos, F. J., Meléndez, J., Gamero, F. I., Colomer, J., and Ortega, J. A. (2003a). Recognition of system behaviours based on temporal series similarity. *Computación y Sistemas*, 7(1):1–16.

Llanos, D., Gamero, F. I., Colomer, J., and Meléndez, J. (2004a). Residual computation using dynamic time warping. Seville, Spain. World Automation Congress.

Llanos, D., Meléndez, J., and Colomer, J. (2004b). Fault diagnosis of dynamic systems reusing cases. In *AVR 2004 Jornada de Recerca en Automàtica, Visió i Robòtica*, Barcelona.

Llanos, D., Meléndez, J., Colomer, J., Mora, J., Corbella, J., and Sánchez, J. (2003b). Abstraction of significant temporal features of voltage sags recorded in a 25kv substation. page 85, Vigo, Spain. ICREPQ'03 International Conference on Renewable Energy and Power Quality.

Llanos, D., Staroswiecki, M., Colomer, J., and Meléndez, J. (2006). Minimising false alarms caused by communication delays in networked systems. IFAC Safeprocess 2006.

Llanos, D., Staroswiecki, M., Colomer, J., and Meléndez, J. (2007). Transmission delays in residual computation. *IET Control Theory & Applications*, 1(5):1471–1476.

López, R. and Plaza, E. (1997). Case-based reasoning: An overview. *AI Communications IOS Press*, 10(1):21–29.

Luck, R. and Ray, A. (1990). An observer-based compensator for distributed delay. *Automatica*, 26(5):903–908.

Ma, C. and Fang, H. (2005). Stability of networked control systems with multi-step delays based on time-division algorithm. *Journal of Control Theory and Applications*, 3(4):404–408.

Main, J., Dillon, T. S., and Shiu, S. C. K. (2001). *Soft computing in case based reasoning*, chapter A tutorial on case based reasoning, pages 1–28. Springer-Verlag, London, UK.

Maquin, D. and Ragot, J. (1991). Comparison of gross errors detection methods on process data. pages 2254–2261, Brigthon, England. 30th Conference on Decision and Control.

Matveev, A. S. and Savkin, A. V. (2003). The problem of state estimation via asynchronous communication channels with irregular transmission times. *IEEE Trans. on Automatic Control*, 48(4):670–676.

Meléndez, J., Macaya, D., Colomer, J., and Llanos, D. (2004a). Symptom based representation for dynamic systems diagnosis.application to electrical power distribution. Madrid, Spain. 7th European Conference on Case-Based Reasoning (ECCBR2004) - 2nd Workshop on applying case-based re.

Meléndez, J., Macaya, D., Llanos, D., Cobos, J., Sánchez, J., and Castro, M. (2004b). Automatic management of voltage sags recorded in a 25kv substation. Barcelona, Spain. ICREPQ'04 International Conference on Renewable Energy and Power Quality.

Meléndez, J., Mora, J., Llanos, D., Ruiz, M., Colomer, J., and Corbella, J. S. . J. (2003). Classification of short duration faults (voltage sags) in transmission and distribution power systems. University of Cambridge (UK). ECC'03, European Control Conference.

Mendes, M., Calado, J., and Sà da Costa, J. (2006). Fault diagnosis system based in agents. pages 427–432. IFAC Safeprocess 2006.

Mendoça, L., Sousa, J., and Sà da Costa, J. (2008). An architecture for fault detection and isolation based on fuzzy methods. *Expert Systems with Applications*, 2008.

Montestruque, L. A. and Antsaklis, P. J. (2003). On the model-based control of networked systems. *Automatica*, 39:1837–1843.

Montestruque, L. A. and Antsaklis, P. J. (2004). Stability of model-based networked control systems withtime-varying transmission times. *IEEE Transactions on Automatic Control*, 49, Issue:1562–1572.

Mora, J., Llanos, D., Meléndez, J., Colomer, J., and Sánchez, J. (2003a). Classification of sags measured in a distribution substation based on qualitative and temporal attributes. Barcelona. CIRED, 17th International Conference on Electricity Distribution.

Mora, J., Llanos, D., Meléndez, J., Colomer, J., Sánchez, J., and Cobella, J. (2003b). Clasificación de huecos de tensión con base en descriptores cualitativos y temporales. Jornada Nacional de Enegía Eléctrica - CIRED.

Mora, J., Meléndez, J., Llanos, D., Colomer, J., Corbella, J., and Sánchez, J. (2003c). Classification of sags measured in a distribution substation using a fuzzy tool. page 89, Vigo, Spain. ICREPQ'03 International Conference on Renewable Energy and Power Quality.

Neidiga, J. and Lunze, J. (2005). Decentralised diagnosis of automata networks. Proceedings of the 16ht IFAC World Congress.

Nikiforov, I., Staroswiecki, M., and Vozel, B. (1996). Duality of analytical redundancy and statistical approach in faultdiagnosis. *Proceedings of the 13th IFAC World Congress*.

Nilsson, J. (1998). *Real-Time Control Systems with Delays*. Doctoral dissertation, Lund Institute of Technology, Lund, Sweden.

Nilsson, J., Bernhardsson, B., and Wittenmark, B. (1988). Stochastic analysis and control of real-time systems with random time delays. *Automatica*, 34:57–64.

Nomikos, P. and MacGregor, J. F. (1994). Monitoring batch process using multiway principle component analysis. *AIChEJ*, 40(8):1361–1375.

Ogata, K. (1997). *Modern Control Engineering*. Prentice Hall, Englewood Cliffs, NJ, 3rd edition.

Patankar, R. (2004). A model for fault-tolerant networked control system using ttp/ccommunication. *IEEE Transactions on VehicularTechnology*, 53, Issue:1461–1467.

Patton, R. J. and Chen, J. (1991). A review of parity space approaches to fault diagnosis. Baden-Baden. Ifac Safeprocess Symposium.

Puig, V., Quevedo, J., Escobet, T., and De las Heras, S. (2002). Robust fault detection approaches using inteval models. Barcelona. IFAC World Congress.

Quinlan, J. R. (1986). Induction of decision trees. *Mach. Learn.*, 1(1):81–106.

Rabiner, L. and Juang, B. (1993). *Fundamentals of Speech Recognition*. Prentice Hall.

Raich, A. and Cinar, A. (1997). Diagnosis of process disturbances by statistical distance and angle measures. *Comput. Chem. Eng.*, 21(6):661–673.

Reiter, R. (1987). A therory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–96.

Reklaitis, G. V., Ravindran, A., and Ragsdell, K. M. (1983). *Engineering Optimization. Methods and Applications*. John Wiley and Sons, USA.

Sakoe, H. and Chiba, S. (1978). Dynamic programmning algorithm optimization for spoken word recognition. *IEE Trans. acoustics, Speech and Signal Proc.*, ASSP-26(1):43..49.

Sankoff, D. and Kruskal, J. (1983). *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison Wesley.

Sauter, D. and Boukhbza, T. (2006). Robustness against unknownnetworked induced delays of observer based FDI. IFAC Safeprocess 2006, pp 331–336.

Sauter, D., Boukhbza, T. and Hamelin, F. (2006). Decentralized and autonomous design for FDI/FTC of networked control systems. IFAC Safeprocess 2006, pp 163–168.

Sauter, D., Mary, N., Sirou, F., and Thieltgen, A. (1994). Fault diagnosis in systems using fuzzy logic. pages 883 – 888. Third IEEE Conference on Control Applications.

Savkin, A. V. and Petersen, I. R. (2003). Set-valued state estimation via a limited capacity communication channel. *IEEE Trans. on Automatic Control*, 48(4):676–680.

Sà da Costa, J., Santos, B., and Mendes, M. (2007). Multi-agent toolbox for fault tolerant networked control systems design. Nancy, France. 3rd International Workshop on Networked Control Systems : Tolerant to Faults.

Shanbin, L., Sauter, D., and Aubrun, C. (2006). Robust fault isolation filter design for networked control systems. pages 681 – 688, Czech Republic. IEEE Conference on Emerging Technologies and Factory Automation. ETFA '06.

Silverman, H. F. and Morgan, D. P. (1990). The application of dynamic programming to connected speech recognition. *IEEE ASSP Magazine*, 7:6–25.

Simoudis, E. (1992). Using case-based retrieval for customer technical support. *IEEE Expert: Intelligent Systems and Their Applications*, 7(5):7–12.

Singhal, A. and Seborg, D. E. (2002). Pattern matching in historical batch data using pca. *IEEE Control Syst.*, 22(5):53–63.

Srinivasan, R., Wang, C., Ho, W., and Lim, K. (2004). Dynamic pca based methodology for clustering process states in agile chemical plants. *Ind. Eng. Chem. Res*, 43(9):2123–2139.

Staroswiecki, M. (2005). Intelligent sensors : A functional view. *IEEE Tr. on Industrial Informatics*, 1(4).

Staroswiecki, M., Cassar, J. P., and Declerck, P. (2000). A structural framework for the design of fdi systems in large scale industrial plants. In Clark, R. N., Patton, R. J., and Frank, P. M., editors, *Issues of Fault Diagnosys for Dynamic Systems*. Springer-Verlag.

Staroswiecki, M. and Comtet-Varga, G. (2001). Analytic redundancy relations for fault detection and isolation inalgebraic dynamic systems. *Automatica*, 37, Issue(5):687–699.

Tipsuwan, Y. and Chow, M. Y. (2003). Control methodologies in networked control systems. *Control Engineering Practice*, 11:1099–1111.

Travé-Massuyès, L. and Dague, P. (2003). *Modèles et raisonnements qualitatifs*.

Travé-Massuyès, L., Escobet, T., Pons, R., and Tornil, S. (2001). The caen diagnosis system and its automatic modelling method. *Computación y sistemas*, 5(2):128–143.

Trevelyan, J. (2004). Industrial process communications and software. Technical report, Mechatronics Design 310, http://www.mech. uwa.edu.au/mechatronics/site-index.html.

Trinh, H. and Aldeen, M. (1997). A memoryless state observer for discrete time-delay systems. *IEEE Trans. on Automatic and Control*, 11(42):1572–1577.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S. H. (2003). A review of process fault detection and diagnosis parts: I, ii, and iii. *Computer and Chemical Enegineering*, 27:293–346.

Villegas, T. and De la Fuente, M. (2006). Fault diagnosis using an observers bank of dynamical neural networks. St. Louis, EEUU. Neural Networks in Engineering ANNIE'06.

Vitturi, S. (2000). Some features of two feildbuses of the iec 61158. *Computer Standards and Interfaces*, 22:203–215.

Walsh, G. C., Beldiman, O., and Bushnell, . G. (2002). Error enconding algorithms for networked control systems. *Automatica*, 38:264–267.

Walsh, G. C. and Hong, Y. (2001). Scheduling of networked control systems. *IEEE Control System Magazine*, pages 57–65.

Wang, Z., Daniel, W. C., and Liu, X. (2003). Variance-constrained filtering for uncertain stochastic systems with missing measurements. *IEEE Trans. on Automatic Control*, 48(7):1254–1258.

Webb, A. R. (2002). *Statsitical Pattern Recognition*. John Wiley and Sons Ltd., 2nd edition.

Wilke, W. and Bergmann, R. (1998). Techniques and knowledge used for adaptation during case-based problem solving. In *IEA/AIE '98: Proceedings of the 11th International Conference on Industrial and Engineering Applications of Artificial In telligence and Expert System*, pages 497–506, London, UK.

Willsky, A. S. (1976). A survey of design methods for failure detection in dynamic systems,. *Automatica*, 12:601–611.

Xie, L., Ji, Z., Pan, T., and Fang, H. (2005). Fault detection for networked control systems with asynchronous measurements delays. *Jounal of System Simulation*, 11(17):2717–2720.

Yang, H. L. and Saif, M. (1998). Observer design and fault diagnosis for state-retarded dynamical systems. *Automatica*, 34(2):217–227.

Ye, H., He, R., Liu, H., and G.Wang (2006). A new approach for fault detection of networked control systems. pages 654–659, Australia. IFAC 14th Symposium on System Identification.

Ye, H., Wang, G., and Ding, S. (2004). A new parity espace approach for fault detection based on stationary wavelet transform. *IEEE Trans. on Automatic Control*, 2(49):281–287.

Ye, H., Zhang, P., Ding, S., and Wang, G. (2000). A time.frequency domain fault detection approach based on parity relation and wavelet transform. pages 4156–4161, Australia. 39th IEEE Conference on Decision and Control.

Yi, B. K., Jagadish, H., and Faloustsos, C. (1998). Efficient retrieval of similar time sequences under time warping. pages 23–27.

Z. Bak, M., Larsen, T. D., Norgaard, M., Andersen, N. A., Puolsen, N. K., and Ravn, O. (1998). Location estimation using delayed measurements. pages 180–185, Coimbra. 5th International Workshop on Advanced Motion Control AMC98.

Zhang, P. and Ding, S. X. (2007). On Monotonicity of a class of optimal fault detection performance versus sampling period. pages 2596–2601, New Orleans, LA, USA,. 46th IEEE Conference on Decision and Control.

Zhang, P., Ding, S. X., Frank, P. M., and Sader, M. (2004). Fault detection of networked control systems with missing measurements. pages 1258–1263, Melbourne, Australia. 5th Asian Control Conference.

Zhang, P., Ye, H., Ding, S. X., Wang, G., and Zhou, D. (2006). On the relationship between parity space and h_2 approaches to fault detection. *Systems & Control Letters*, 2(55):91–100.

Zhang, W., Branicky, M. S., and Phillips, S. M. (2001). Stability of networked control systems. *IEEE Control Systems Magazine*, pages 84–99.

Zheng, Y., Fang, H., and Wang, H. (2004). Kalman filter based fault detection of networked control system. pages 1330–1333, China. 4th World Congress on Intelligent Control and Automation.

Zheng, Y., Fang, H., Wang, H., and Li, L. (2003a). Observer-based fdi design of networked control system with output transfer delay. *Control Theory and Applications*, 5(20):653–663.

Zheng, Y., Fang, H., Wang, H., and Xie, L. (2003b). Fault detection approach for networked control system based on a memoryless reduced-order observer. *ACTA Automatica SINICA*, 4(29):559–566.

Zheng, Y., Hua, X., Fang, H., and Wang, H. (2005). Research on observed-based fault diagnosis of networked control system. *Systems Engineering and Electronics*, 6(27):1069–1072.

Zhivoglydov, P. V. and Middleton, R. H. (2003). Networked control design for linear systems. *Automatica*, pages 743–750.

Zhong, M., Ye, H., Wang, G., and Zhou, D. (2005). Fault detection filter for linear time delay systems. *Nonlinear Dynamics and Systems Theory*, 5(3):273–284.

Zhong, M. S., Ding, S. X., and Lam, J. (2004). An optimization approach to fault detection observer design for neutral delay systems. *Dynamics of Continuous, Discrete and Impulsive Systems (B)*, 11(6):701–721.