ISBN: 978-972-9059-05-6

the origin/la source : http://www.afscet.asso.fr/resSystemica/Lisboa08/britopalma.pdf
the title/le titre : **Dealing with complexity in supervision systems.**
the author/l'auteur : **BRITO PALMA Luís, COITO Fernando Vieira, Paulo Sousa GIL & Hermínio DUARTE-RAMOS**
the pages/la pagination : **9 p.**
the year/l'année : **2008**
& the book/la publication: **7th Systems Science European Union Congress Proceedings, Lisboa, Portugal.**

Systems Science European Union
Union Européenne de Systémique
UES

APOCOSIS

7th Congress of the UES
Systems Science European Union
Lisbon
Dec 17-19, 2008

Dealing with complexity in supervision systems

Luís Brito Palma
Paulo Sousa Gil

Universidade Nova de Lisboa (FCT)
Electrical Engineering Department
{lbp, psg}@fct.unl.pt

Fernando Vieira Coito
Hermínio Duarte-Ramos

Universidade Nova de Lisboa (FCT)
Electrical Engineering Department
{fjvc, hdr}@fct.unl.pt

Abstract

Industrial plants are becoming increasingly complex, generating huge amount of process data and requiring real-time supervision. In the last two decades, in part due to the use of high efficient dedicated computers and advanced instrumentation, the monitoring and supervision of industrial systems have allowed not only high quality performances but also a wide understanding of the underlying plant's behaviour. The huge amount of streaming data requires efficient data mining approaches. When the system is complex, or difficult to model, black-box nonlinear identification is usually a good and often the only alternative. Based on the available information, it is necessary to extract knowledge from the process, in order to perform many important tasks like monitoring, fault detection and diagnosis, control and supervision.
Most supervision systems can be classified as complex systems, since they are composed of interconnected modules that, as a whole, exhibit behaviours not obvious from the properties of each individual component.
The main contributions of this paper are focused on new systemic approaches to deal with complexity in supervision systems. Architectures and key concepts will be presented and the role of computational intelligence techniques such as fuzzy logic and neural networks based approaches together with optimization and dimensional/complexity reduction techniques will be highlighted.

Keywords: complexity, supervision systems, nonlinear models, dimensional reduction techniques, computational intelligence, optimization.

Introduction

Nowadays, supervision systems play an important role in industry mainly due to the increasing demand for product quality and high efficiency, and to the growing integration of automatic control systems in technical processes.

The supervision of technical processes shows the present state, indicating undesired or not permitted working regions, and taking appropriate actions to avoid damage or accidents. The main idea is to guarantee that faults do not cause drastic failures.

By going through the literature, one recognizes immediately that the terminology in this field is not consensual. Some definitions are presented here accordingly to Isermann & Balle [8]. A fault is a non permitted deviation of at least one characteristic property or parameter of the system from the acceptable/usual/standard condition. A failure can be defined as a permanent interruption of a system's ability to perform a required function under specified operating conditions.

A supervision system has a great number of components and interconnections, and it is difficult to describe and understand its behaviour, so it can be classified as a complex system. A hard question is: How to deal with complexity in supervision systems? The response is not obvious or straightforward. Systemic approaches to deal with complexity in supervision systems are proposed in this paper.

The paper has the following sections: introduction, complex systems, supervision systems, methodology to deal with complexity in dynamic systems and conclusions.

Complex systems

Clearly the notions of system and of signal are broad concepts, and it is not surprising that they play an important role in modern science [11].

A signal is a function of one or more independent variables, and typically contains information about the behaviour or nature of some phenomenon [13]. A system responds to particular input signals by producing other output signals [13]. Another definition is: a group of interacting, interrelated or interdependent elements forming a complex whole. In loose terms, a system is an object in which variables of different physical nature interact and produce observable signals [11].

Usually, the interesting observable signals are aggregated in a output signal $\mathbf{y}(k)$. Here k represents a discrete time. The system is also affected by external signals. The external signals that can be manipulated by the operator are called input signals $\mathbf{u}(k)$. Others are called disturbances, and can be divided into those that are directly measured $\mathbf{w}(k)$ and those only observed through their influence on the system output $\mathbf{v}(k)$ (Figure 1).

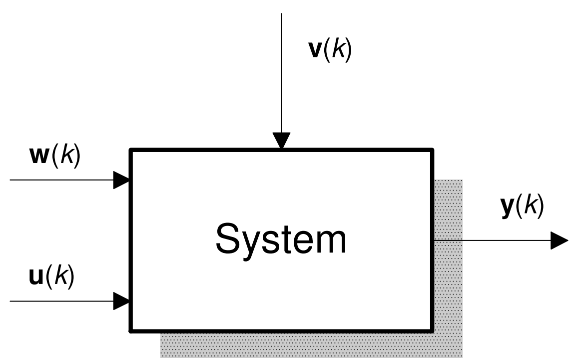

Figure 1: A system with inputs, disturbances and outputs.

In fact, most of the dynamic systems in our world are complex systems. A great famous quote about Complex Systems comes from Aristotle, the Greek philosopher, who said: "The whole is more than the sum of its parts". This is actually a holistic perspective of systems.

One possible definition for a complex system is proposed here: a dynamic system, with various components and interconnections, that is difficult to describe, understand its behaviour, model and predict, design, control, and supervise. Several other definitions for a complex system can be found in literature, being most of them focused on the intrinsic nonlinear behaviour of systems. Another definition of a complex system is [9]: a high-dimensional nonlinear system, which can be (but not necessarily is) adaptive. In nature, nothing is linear. In reality, everything has a certain degree of nonlinearity, which means a certain degree of unpredictability and uncontrollability. Yet another definition [17]: a complex system is composed of a large number of elementary agents which carry various states and interact under nonlinear function.

The human being is one of the most complex systems existing in our world. Within our body, the human mind is certainly a high complex system. According to Ackoff [1], the content of the human mind can be classified into five categories: data, information, knowledge, understanding and wisdom.

In our world, there exist a great number of other systems that can be classified as complex systems. Societies are complex dynamic systems where synergy and emergence play a crucial role. Other complex systems are biological systems, ant colonies, economy, climate, nervous systems, modern energy or telecommunication infrastructures, etc.

Supervision systems

The supervision systems in industrial plants, implemented in SCADA software, must undertake, at least, the following three main tasks: monitoring, control and fault tolerance.

Monitoring generally means to be aware of the state of a system. This task involves acquiring and evaluating the behaviour of input and output signals, and possibly the estimation of certain process parameters.

Control focuses on the modelling of systems, on analyzing their dynamic behaviour, and using control theory to design a controller to guarantee that the systems have a desired dynamic behaviour in closed loop.

Fault tolerance implies compensation of fault effects in such a way that they do not provoke the system's failure. To reach this goal it is necessary to implement redundancy (in hardware or software). The main task to be tackled in achieving fault tolerance is the design of a controller with suitable structure to guarantee stability and satisfactory performance, in the case of faults on process components, sensors or actuators [14]. The fault tolerant control system must possess integrity in the control loops.

Fault tolerance in automatic control systems is gaining more and more importance, and can be achieved either by passive or by active strategies [2, 14].

Passive strategies are centred mainly on robust control. A closed-loop system can have some degree of fault-tolerance by means of a carefully chosen feedback design, taking care on effects of both faults and system uncertainties.

Active strategies on the other hand imply intelligent control, system identification, fault detection and diagnosis, reconfiguration (i.e. new controller parameters) or restructuring (i.e. the system structure itself can be changed). Active fault-tolerant systems based on unanticipated faults must have a mechanism for identifying abnormal system changes. This is essentially the function of a fault detection and diagnosis (FDD) scheme.

Fault tolerance approaches need the implementation of fault detection and diagnosis (FDD) methods. In Figure 2, a typical model-based FDD architecture is depicted. The fault detection (FDE) task concerns the determination of the faults present in a system, and the time of detection. The fault diagnosis (FDG) task includes the fault isolation (i.e., determination of the kind, location and time of detection of a fault) and fault identification (i.e., determination of the size and time-variant behaviour of a fault).

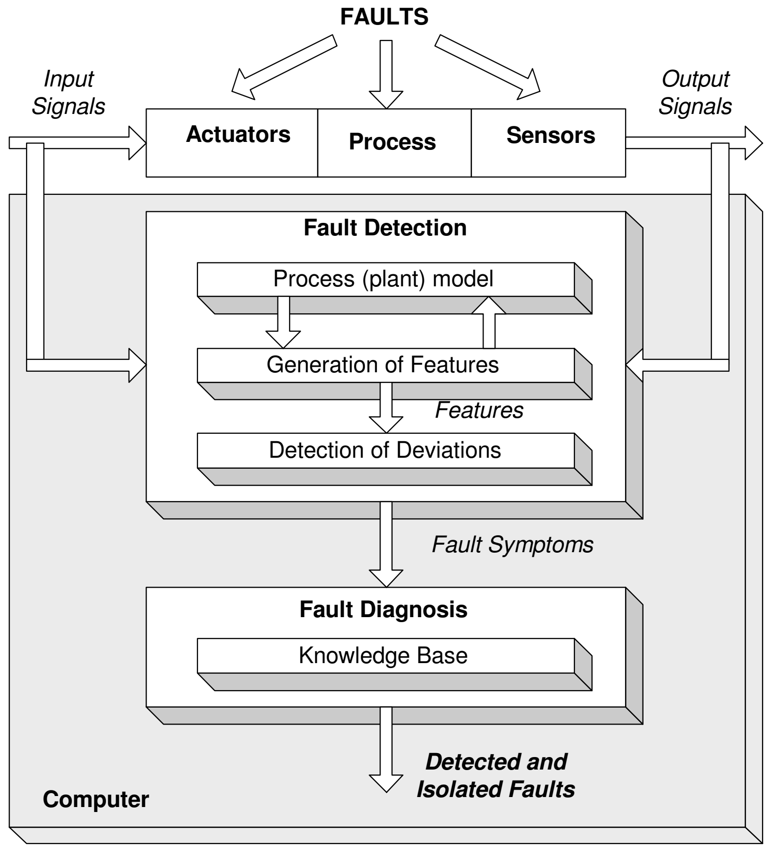

Figure 2: Typical model-based fault detection and diagnosis architecture.

Depending on the particular approach for features (residuals, parameter deviations, etc) generation, the methods of fault detection can be divided into three main categories, as depicted in Figure 3, [3]: hardware redundancy, model-based and signal/data based. The main sub-categories are also shown. Computational intelligence techniques, mainly neural networks and fuzzy logic, play a crucial role in many approaches. Principal component analysis (PCA) is also a method widely used [3]. PCA is one of the most popular dimensionality reduction techniques. It is a multivariate statistical technique in which a number of related variables are transformed to a smaller set of uncorrelated variables.

Fault diagnosis methods can be classified into two main categories [8]: classification methods, and reasoning methods.

If several symptoms change differently for certain faults, one way to determine the presence of a fault is to use classification methods which evaluate indicate changes on symptom vectors. Some classification methods are: a) geometrical distance and probabilistic methods; b) artificial neural networks; c) fuzzy clustering.

If more information about the relations between symptoms and faults is available in the form of diagnostic models, reasoning methods can be applied. Diagnostic models can be expressed in the form of symptom-fault causality. The causalities events can be expressed as “if-then” rules. Then analytical as well as heuristic symptoms (from operators) can be processed. By considering them as inaccurate facts, probabilistic or fuzzy-set descriptions lead to a unified symptom representation. By forward and backward reasoning, probabilities or possibilities of faults are obtained as a result of diagnosis. Typical approximate reasoning methods include: a) probabilistic reasoning; b) possibilistic reasoning with fuzzy logic; c) reasoning with artificial neural networks.

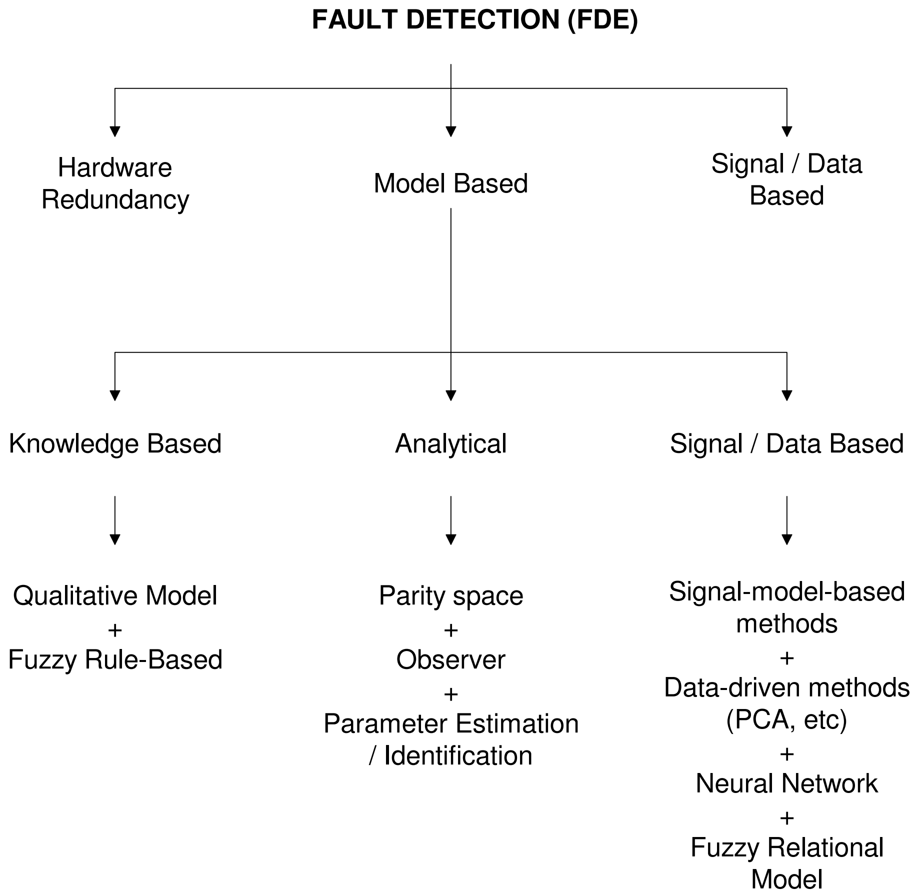

Figure 3: Scheme of features (residuals, parameter deviations, etc) generation approaches.

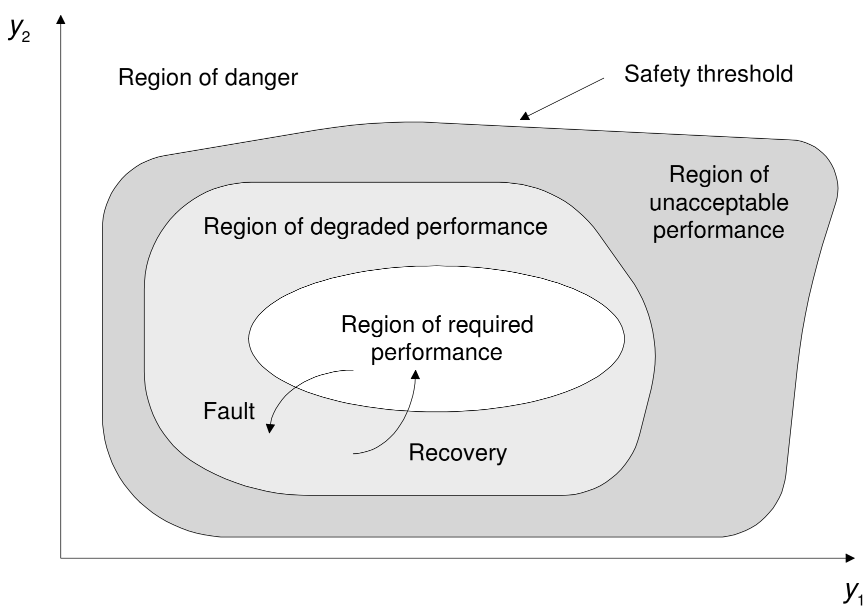

Figure 4: Performance regions.

Due to its great importance, the relationship between fault tolerance and safety must be detailed. Assuming that the system performance can be described by two variables y_1 and y_2, in a two dimensional space, then different regions have to be considered as depicted in Figure 4. During nominal operation, the controller must guarantee that the system remains in the region of required performance. In some situations, mainly when small faults occur, the controller "hides" the effects of faults, and this makes the detection and diagnosis tasks more difficult. A fault induces the system to move to the region of degraded performance, and the fault tolerant controller (FTC) should be able to carry out or trigger recovery actions. A safety system must interrupt the operation of the overall system so as to avoid critical degradation or danger, if the performance reaches the safety threshold. A fault tolerant controller and the safety system work in different regions of the signal space, they are usually implemented in separate units, and that allows an independent design.

Tiny faults are difficult to detect but easy to correct, whereas severe faults are easy to identify but normally difficult to compensate.

Modern technological systems consist of several, often many subsystems, which are strongly interconnected. The effect of a fault in a single component is usually propagated throughout the overall system, as depicted in Figure 5. If a fault determines the safety system to shut off the whole system, then the fault has caused a system failure.

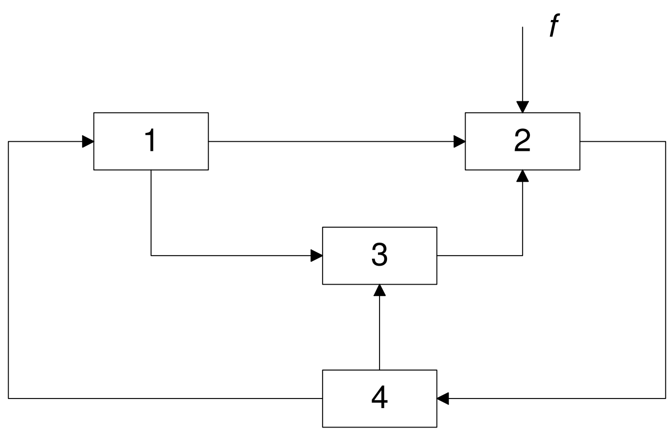

Figure 5: Fault propagation in interconnected systems.

Most of the subsystems in industrial plants are, typically, nonlinear MIMO systems, i.e., multi-input multi-output signals with nonlinear behaviour. Models are needed for controller design and for fault detection and diagnosis (FDD). Modelling this kind of industrial plants require the use of nonlinear models. From a theoretical point of view, a white-box model is more desirable to perform control and the FDD tasks, but in most cases it is very hard, or even impossible, to obtain it. When the systems are complex, or difficult to model, modelling based on black-box models is usually a good and often the only alternative.

Nonlinear black-box models can be constructed using multi-model approaches, neural networks [3, 6], fuzzy logic [4], etc. The estimation of the model parameters implies the use of optimization techniques.

There exist many approaches to optimization [5, 15, 16]. Some require the existence of differentiable functions, other do not need this requirement.

Assuming that the function to be minimized (objective function) is differentiable then some classical optimization methods can be applied, depending on the kind of the underlying objective function and variable types: linear programming, quadratic programming, nonlinear programming, stochastic programming, etc.

For the case of non-differentiable objective function it is necessary to apply search methods such as on genetic algorithms [12], ant colony optimization or particle swarm optimization [5], etc.

The role of the human being within the supervisions systems must also be considered [7].

In Figure 6, a typical fault tolerant control (FTC) architecture is depicted [14]. The thick lines represent signal flow, and the thin lines represent adaptation (tuning, scheduling or reconfiguration). The supervision system plays a crucial role in FTC applications. The supervisor must take decisions about adaptation when faults occur, in order to maintain the desired system performance and preserve the stability of the overall system.

In some non-severe fault cases, the supervisor only needs to perform the re-tuning of the controller. However, when severe structural fault occurs, the supervisor usually needs to change the control strategy using other sensors, actuators, re-tuning the controllers and also changing the set-points. In most critical situations, the final decisions are taken by the humans.

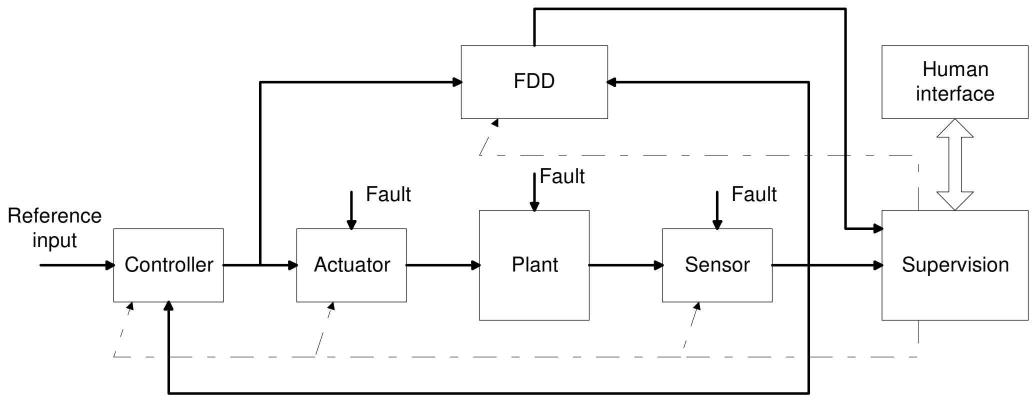

Figure 6: Fault tolerant control architecture.

Methodology to deal with complexity in dynamic systems

The ideas and key concepts described above, related to a supervision system of an industrial plant, can be applied directly to a general complex dynamic system. Each complex system needs to be under supervision, i.e., to be monitored, controlled and to be fault tolerant.

Here, it is formulated a systemic approach to deal with complex dynamic systems. The overall architecture, including the key concepts and the interconnections, to deal with complex dynamic systems can be observed in Figure 7. Only signal flows are depicted.

The main idea is to develop a supervision system to interact with the complex dynamic system. The supervision system makes use of nonlinear models, dimensional reduction techniques, computational intelligence, optimization and other techniques.

Nonlinear models are needed to model the nonlinear behaviour of the system, to design controllers and to develop methodologies of fault detection and diagnosis. Such models may be ARX multi-models and models based on computational intelligence (neural models and fuzzy models), among others.

Dimensional reduction techniques like principal components analysis (PCA) allow reducing the number of signals to be analysed, enabling a better interpretation by human beings, as they typically can only understand well two-dimensional plots.

Computational intelligence can be used for decision making, and also for nonlinear modelling, control, pattern recognition, fault detection and diagnosis, etc.

Others techniques may be necessary to deal with complex systems, like signal processing, multivariate statistics, etc.

Optimization is needed for almost everything such as to estimate system parameters, to design a controller, to train a neural network, to design a fuzzy model or controller, etc.

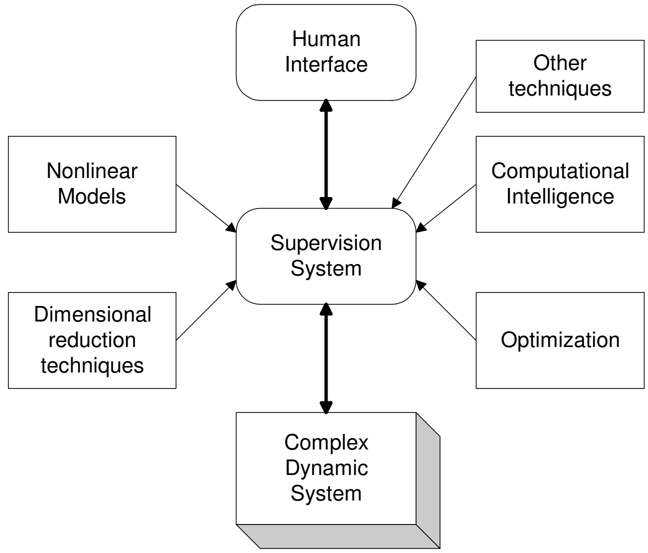

Figure 7: Systemic architecture to deal with complex dynamic systems.

Conclusions

This paper presents a systemic approach to deal with complex systems. This approach incorporates a supervision system, to supervise the complex system that is build up by integrating nonlinear models, dimensional reduction techniques, computational intelligence, optimization and other techniques.

Systemics is the science of systems. To understand our world we need to think that almost everything can be regarded as a system, where signals must be monitored, controlled and ultimately supervised. The supervision of systems should by nature try to guarantee the observability, the controllability and, most important, the system stability.

The architectures and key concepts presented can be applied to any complex system like a social system, an economic system, a biological system, etc. This extrapolation is a pointer for future research.

Acknowledgements

The authors wish to thank the support of Faculdade de Ciências e Tecnologia of Universidade Nova de Lisboa for the current research project.

References

[1] Ackoff, R., «From Data to Wisdom», Journal of Applied Systems Analysis, 1989, vol. 16, pp. 3-9.
[2] Blanke, M., M. Kinnaert, J. Lunze, M. Staroswiecki, Diagnosis and Fault-Tolerant Control, Springer, 2003.
[3] Brito Palma, L., Fault Detection, Diagnosis and Fault Tolerance Approaches in Dynamic Systems based on Black-Box Models, , PhD thesis, Universidade Nova de Lisboa – FCT – DEE, 2007.
[4] Driankov, D., H. Hellendoorn, M. Reinfrank, An Introduction to Fuzzy Control, Springer-Verlag, 1996.
[5] Engelbrecht, A., Fundamentals of Computational Swarm Intelligence, John Wiley & Sons, 2005
[6] Hagan, M., H. Demuth, M. Beale, Neural Network Design, PHS Publishing Co., 1995.
[7] Isermann, R., Fault-Diagnosis Systems - An Introduction from Fault Detection to Fault Tolerance, Springer, 2006.

[8] Isermann, R., P. Balle, «Trends in the Application of Model-Based Fault Detection and Diagnosis of Technical Processes», Control Eng. Practice, 1997, vol. 5, no. 5, pp. 709-719.
[9] Ivancevic, V., T. Ivancevic, Geometrical Dynamics of Complex Systems, Springer, 2006.
[10] Levine, W. (editor), The Control Handbook, CRC Press, 1996.
[11] Ljung, L., System Identification - Theory for the User, Prentice-Hall, 1999.
[12] Man, K., K. Tang, S. Kwong, Genetic Algorithms - Concepts and Designs, Springer, 1999.
[13] Oppenheim, A., A. Willsky, Signals & Systems, Prentice-Hall, 1983.
[14] Patton, R., «Fault Tolerant Control Systems: the 1997 Situation», Proc. of the Safeprocess Symposium, Hull, UK, 1997.
[15] Pedregal, P., Introduction to Optimization, Springer, 2004.
[16] Ruszczynski, A., Nonlinear Optimization, Princeton University Press, 2006.
[17] Yu, N., M. Li, X. Ruan,«Applications of Cellular Automata in Complex System Study», Int. Journal of Information and Systems Sciences, 2005, vol. 1, pp. 302-310.