



HAL
open science

Online Multiagent Monitoring in Intensive Care Units

Kaouther Noura

► **To cite this version:**

Kaouther Noura. Online Multiagent Monitoring in Intensive Care Units. Multiagent Systems [cs.MA]. Université de Tunis (Tunisie), 2010. English. NNT : . tel-03543241

HAL Id: tel-03543241

<https://theses.hal.science/tel-03543241>

Submitted on 25 Jan 2022

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

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



Online Multi-Agent Monitoring in Intensive Care Units

THESE

***En vue de l'obtention du Doctorat
En Informatique de Gestion***

Présentée et soutenue publiquement par:

Mme Kaouther NOUIRA-FERCHICHI

Le 23 Juin 2010

Membres du Jury:

- **M. Khaled MELLOULI** : Professeur à l'Université de 7nov. Carthage : Président
- **M. Abdelwahed TRABELSI** : Professeur à l'Université de Tunis : Directeur de thèse
- **Mme Nahla BEN AMOR** : Maître de Conf. à l'Université de Tunis : Rapporteur
- **Mme Saoussen KRICHEN** : Maître de Conf. à l'Université de Tunis : Rapporteur
- **M. Mohamed AYADI** : Professeur à l'Université de Tunis : Membre

To my father Abdelhamid, my mother Nebiha.

To my husband Ahmed, my daughters Molka and Olfa.

To my brother Yassine, his wife Wafa, his daughter Ibtissem.

To my brother Kais, his wife Amina, his daughter Mariem, his son Mohamed Karim.

ACKNOWLEDGEMENTS

I would like to thank the following parties: my supervisor, my family, my colleagues, and the thesis committee.

My supervisor Professor **Abdelwahed Trabelsi**, helped me to specify the research topic, and guided me throughout this research.

My family members: **Ahmed** (my husband), **Molka** and **Oifa** (my daughters), supported me in the course of the thesis period.

My colleague **Khaled Omrani** at ISSATso (Institut Supérieur des Sciences Appliquées et des Technologies de Sousse), helped me to manage my teaching schedules many times.

The thesis committee honoured me to comment on this thesis, and to accept to discuss its funds.

Contents

List of Figures	4
List of Tables	6
1 INTRODUCTION	7
1.1 Motivation	7
1.2 The Subject of the research	10
1.2.1 Systems theory based architecture	13
1.2.2 Agent based system	14
1.2.3 Signal processing and forecasting	15
1.3 Some definitions	16
1.4 Learning outcomes of the thesis	17
1.5 Structure of this thesis	19
2 MONITORING ISSUES IN INTENSIVE CARE UNITS	20
2.1 Introduction	20
2.2 Patient monitoring systems issues	23
2.3 Attempts to reduce false alarms using statistics	24
2.4 Attempts to reduce false alarms using artificial intelligence	25
2.5 Attempts to reduce false alarms using standardization	26
2.6 Conclusion	28
3 TIME SERIES ANALYSIS IN MEDICAL DATA	29
3.1 Introduction	29
3.2 Outlier detection	31
3.3 Level change detection	35
3.4 Change point detection	38
3.5 Models for on-line monitoring and forecasting medical time series	38
3.6 Multivariate time series and dimension reduction	40
3.7 Conclusion	43
4 ONLINE MULTIAGENT MONITORING SYSTEM FOR INTENSIVE CARE UNITS	44
4.1 Introduction	44
4.2 System description	45
4.2.1 The Acquisition System Architecture	45
4.2.2 The Monitoring System Architecture	46

4.2.3	The Alarm System Architecture	48
4.3	The System's working principles	48
4.4	Conclusion	53
5	EXPERIMENTAL STUDY	56
5.1	Introduction	56
5.2	Control Agent functionality	57
5.3	The ARIMA model	58
5.3.1	Autoregressive (AR) process	58
5.3.2	Moving Average (MA) process	60
5.3.3	Integration	60
5.3.4	Seasonality	61
5.3.5	Building ARIMA models	61
5.4	Automation of the modeling steps	62
5.5	Anomaly detection	63
5.5.1	Outliers' detection rule	63
5.5.2	Level change, trends and variance change detection rules	64
5.6	Alarm Agent functionality	64
5.7	Data description	64
5.8	Experimental topology	64
5.9	First set of experiments	65
5.9.1	Patient 1	65
5.9.2	Patient 2	67
5.10	Second set of experiments	69
5.11	Conclusion	70
6	CONCLUSIONS AND FUTURE RESEARCH	71
6.1	Conclusions	71
6.2	Future research	73
A	MAMS ANALYSIS AND DESIGN WITH AUML	75
A.1	System Specification	75
A.1.1	Actors Model	75
A.1.2	Goals definition	75
A.1.3	Roles definition	78
A.1.4	Use case model	80
A.1.5	MAMS use case diagram	80
A.2	System Analysis	88
	Bibliography	93

List of Figures

1.1	Level changes	8
1.2	Variance changes	8
1.3	Graphical representation of variables interactions of non critical state [39]	10
1.4	Graphical representation of variables interactions of pulmonary hypertension state [39]	11
1.5	Graphical representation of variables interactions of congestive heart failure state [39]	11
1.6	Graphical representation of variables interactions of Vasopressure support state [39]	12
1.7	Multivariate Time Series	12
1.8	Hierarchy and interaction between systems	14
1.9	Systems interact with the external environment through Sensors and Deliverers	15
2.1	Data Flow Diagram of Alarm System in Intensive Care Monitoring System	23
2.2	MetaMonitor devices. Source [29]	28
3.1	Outliers give false alert and mask changepoints	30
3.2	Undirected edge	41
3.3	Directed edge	41
4.1	Global MAMS Architecture	45
4.2	The Acquisition System Architecture (a)	46
4.3	The Acquisition System Architecture (b)	47
4.4	The Monitoring System Architecture (a)	47
4.5	The Monitoring System Architecture (b)	48
4.6	The Alarm System Architecture (a)	49
4.7	The Alarm System Architecture (b)	49
5.1	ARIMA-model building procedure	62
5.2	Subset of Patient 1 variables	65
5.3	Subset of patient 2 variables' values	67
6.1	Comparison between purely java program and a java program calling R method	73
A.1	MAMS Actors' Model	76
A.2	The tree of goals	76
A.3	Acquisition Agent goal diagram	78
A.4	Control Agent goal diagram	79
A.5	The graphical Agent goal diagram	79

A.6 Alarm Agent goal diagram	80
A.7 MAMS use case diagram	81
A.8 System activation use case diagram	81
A.9 Acquisition System Use Case Diagram	82
A.10 Acquisition Agents and monitoring system interaction use case diagram	82
A.11 Control Agent use case diagram	83
A.12 detailed Control Agent use case	84
A.13 Control and Graphical Agents interaction use case diagram	84
A.14 Graphical Agent use case diagram	85
A.15 Intraction between Alarm system and Control Agents as extern actors (a)	85
A.16 Intraction between Alarm system and Control Agents as extern actors (b)	86
A.17 MAMS turn off use case diagram	86
A.18 Global use case diagram and the interaction between agents	87
A.19 MAMS global activity diagram	87
A.20 Global class diagram	88
A.21 Agent creation sequence diagram	89
A.22 Information reception from Acquisition Agent sequence diagram	89
A.23 Values treatment sequence diagram	90
A.24 Change graphic color sequence diagram	91
A.25 Alarm generation sequence diagram	91
A.26 RAZ request sequence diagram	92

List of Tables

2.1	Tsien classification of alarms [78]	24
2.2	Comparison between single parameter boundary-based module and trend-based module	26
2.3	Significance of alarm level priority	27
3.1	Results of segments analysing	40
5.1	Patient 1 data monitored by Time Series based Control Agents	66
5.2	Patient 1 data monitored by Threshold based Control Agents	66
5.3	Patient 2 data monitored by Time Series based Control Agents	68
5.4	Patient 2 data monitored by Threshold based Control Agents	68
5.5	Systems outputs using Patient 1 data	69
5.6	Systems outputs using Patient 2 data	70

Chapter 1

INTRODUCTION

This chapter introduces the thesis. It defines the learning outcomes and measures of achievements. Along with this, the motivation of the work is defined, and a summary of the contribution of the research work. The chapter also includes background section on monitoring systems in Intensive Care Units (ICUs) and typical systems weak points. Finally, the chapter outlines the structure of the thesis.

1.1 Motivation

In ICUs a large number of variables relevant to patient monitoring like physiological variables, laboratory data, device parameters, etc. [4] must be controlled by the physician. So, for one patient, a physician is confronted to a flood of recorded variables during his typical morning round. This high dimensional data exceed the human ability to identify and develop a solution considering all the observations [32]. As Miller in [57], even a senior physician is unable to provide a systematic solution to any problem involving more than seven variables. Furthermore, human ability to identify the degree of relatedness is limited to only two variables [48].

The use of clinical information systems (CIS) is unavoidable. On the one hand, because they allow real time improved acquisition and storage of the electronic patient records and analyze them in order to improve the quality of therapy and care. On the other hand, they provide physicians with all information they need at the bedside computers. Such computers are so-called on-line monitoring systems.

Nowadays, most of on-line monitoring systems, which are widely used in practice, are based on fixed thresholds alarm systems [33]. This kind of systems produce about 86% of false alarms [81] this is mainly due to measurement artifacts such as patient movements [32] or transient fluctuations [5].

To allow for more reliable approach to intensive care therapy and on-line bedside decision support, intelligent alarm systems have been suggested [44]. These systems' principle is to detect automatically and on-line critical states from high dimensional data. Therefore, new methodological approaches are suggested with quite different tools like a time series analysis, a statistical and automatic process

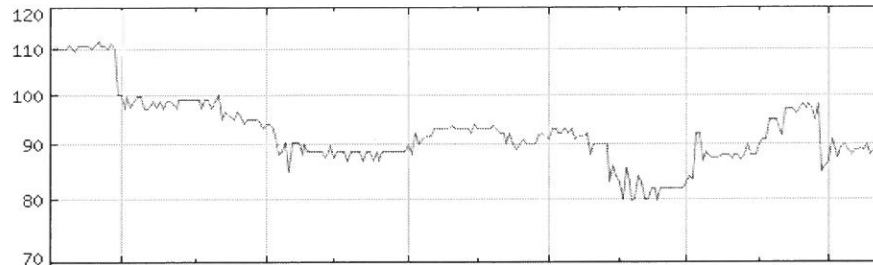


Figure 1.1: Level changes

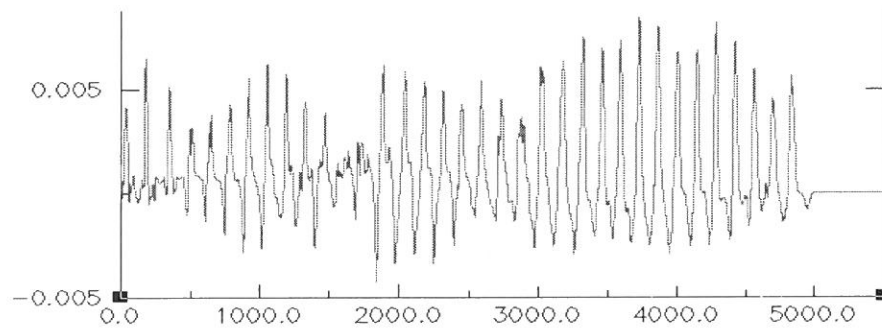


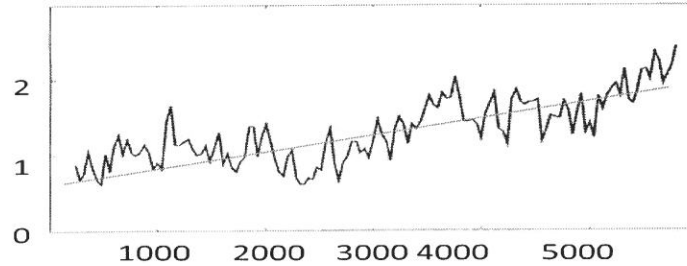
Figure 1.2: Variance changes

control, neural networks, system theory, digital signal processing, artificial intelligence, fuzzy control [5].

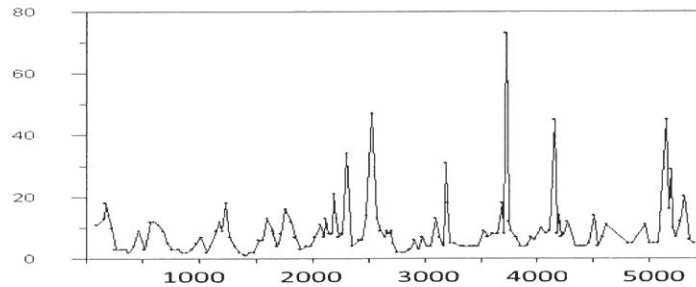
Most data in CIS come in the form of time series which is a set of repeated observations of the same physiological variable X , such as Electrocardiogram or blood pressure, where

$$X = \{x_t\}, t = 1, 2, \dots, T \quad (1.1)$$

and x_t is a random variable. Hence, time Series employed for automatic on-line monitoring systems gives an option for more reliable evaluation of the patient state in ICUs. Because, time-series consists of interesting models for the joint distribution of X . The models impose structure, which we must evaluate to see if it captures and on-line detects the features we think are present in the data such as level changes (see fig. 1.1), variance changes (see fig. 1.2), trends (see fig. 1.1) and outliers (see fig. 1.1).



Trends



Outliers

Actually, in critical care, a multitude of variables are measured in the course of time (e.g. heart rate (HR), Arterial Blood Pressure (ABP), pulmonary Arterial Pressure (PAP), Central Venous Pressure (CVP) Blood Temperature (BTemp), pulsoximetry or Arterial hemoglobin oxygen saturation (SpO₂), ...). Each variable presents a time series. But at any time t , if we consider a vector $x_t = (x_{1t}, x_{2t}, \dots, x_{nt})'$ where each x_{it} , $i = 1, \dots, n$ represents the observation of the variable x_i at time t , we can observe some interactions between its parameters. To exemplify the concept, we consider distinct clinical states already presented in [39]. Pulmonary hypertension, septic shock, congestive hart failure and vasopressor support. Those states are accompanied by different pathophysiological aspects of the circulation system. those aspects may be the result of differences in the interaction between vital signs. In normal state there are strong associations between distinct kinds of blood pressure(ABP, PAP, CVP) and weak associations between each of them and the hart rate (HR). In contrast, no association is supposed between SpO₂ or BTemp and any of the other variables (see fig. 1.3). Giving the example of pulmonary hypertension. This clinical state is characterized by an elevated PAP. Since CVP also influences the right ventricle, one expects strong interactions between CVP and PAP. On the other hand, the higher resistance within the pulmonary bloodstream diminishes the interactions between PAP and ABP. As changes in ABP will have a less than normal effect on CVP. And a weak interaction between SpO₂ and BTemp with the other

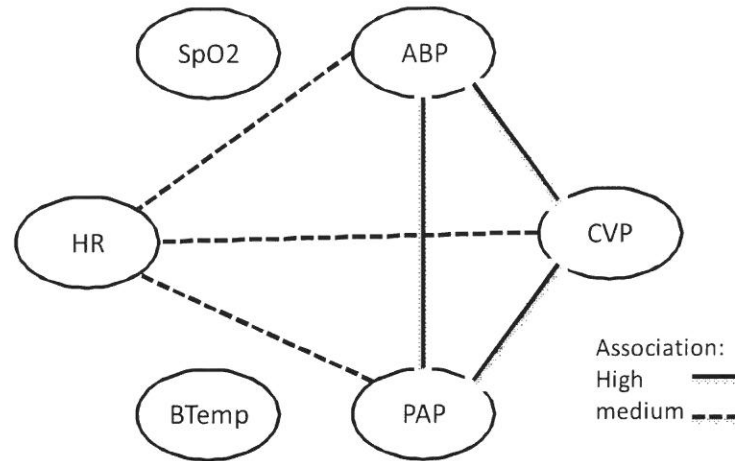


Figure 1.3: Graphical representation of variables interactions of non critical state [39]

variables (see fig. 1.4). In consequence, the expected associations of vital signs show another picture for the state of pulmonary hypertension than the associations under normal physiological conditions. The same thing with the congestive heart failure (see fig 1.5) and the vasopressor support (see fig.1.6). We can conclude that the medical data is not a set of univariate time series working separately, but there are a lot of interactions between them and we have to treat them as a multivariate time series with interdependent components (see fig. 1.7).

Much research works have gone into the development of Multivariate Time Series (MTS) analysis. Such works can be divided into two main classes: statistical approach and Artificial Intelligence (AI) approach [54]. Statistical MTS modelling methods include the Vector Auto-Regressive (VAR) process [85], graphical models [20], [39], [2], and dimension reduction [7], [4]. Such approaches have several limits, in particular, they need a lot of memory and time execution. So, they are not appropriate to develop online systems. Therefore, they can't be suggested for a monitoring system which must be a real time one. AI methods have been developed in the purpose of avoiding statistical MTS modelling methods complexity. They aim to resolve interaction detection in MTS variables [67], [54]. So an adequate intelligent monitoring system must be achieved by combining some time series techniques with methods of artificial intelligence [58].

1.2 The Subject of the research

Our research consists of evaluating current monitoring systems and tries to propose new way of dealing with this problem. The authentication problem described in chapter 2 has directed implication to the bedside decision in ICUs. The research aims to combine time series techniques with intelligent

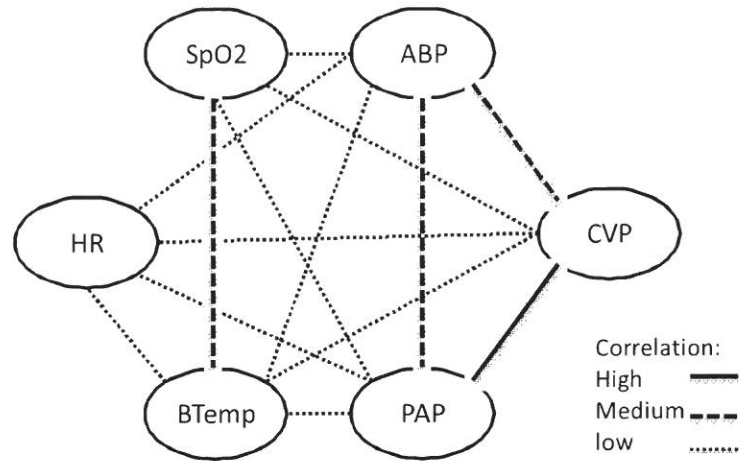


Figure 1.4: Graphical representation of variables interactions of pulmonary hypertension state [39]

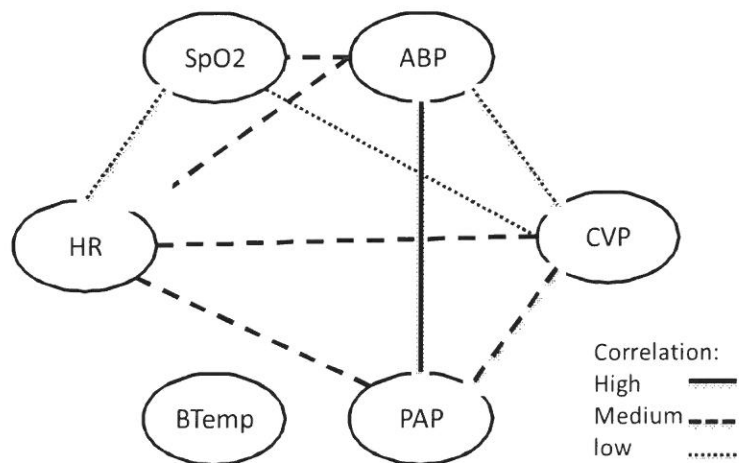


Figure 1.5: Graphical representation of variables interactions of congestive heart failure state [39]

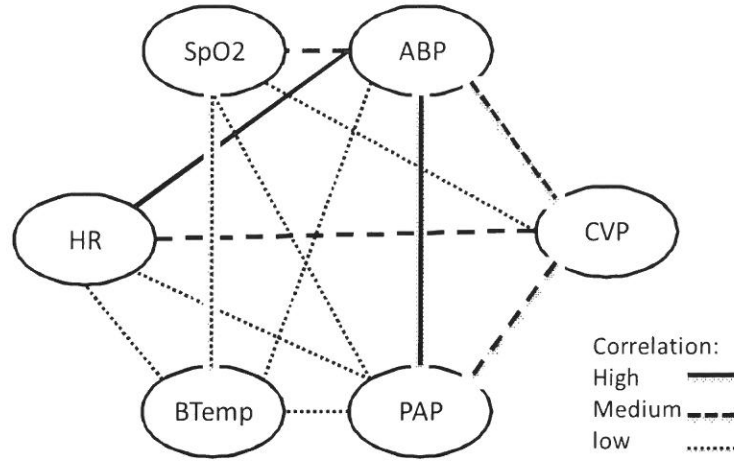


Figure 1.6: Graphical representation of variables interactions of Vasopressure support state [39]

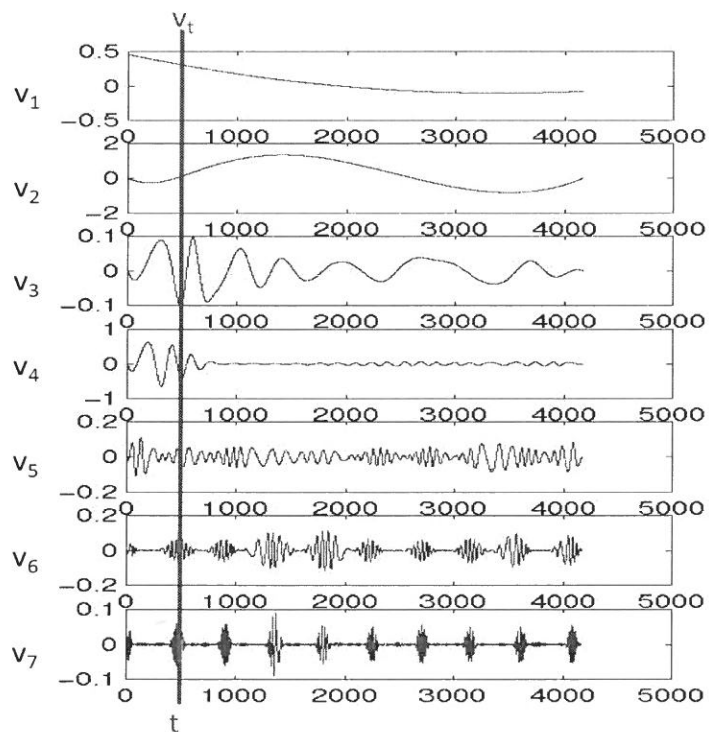


Figure 1.7: Multivariate Time Series

agents. Instead of using production rules, the agents will use statistical evaluation techniques to detect anomalies. When anomaly is detected, the agent sends a request as a message to the supervisor agent which has its own production rules base that provides it the guidance to know the physiological variables interdependence and so which agents' requests must it be waiting before generating alarm. Time series analysis are used to evaluate physiological variable features and forecast its future behavior.

Our research is based on three important parameters:

1. Systems theory based architecture,
2. Agent based theory,
3. Signal processing and forecasting.

1.2.1 Systems theory based architecture

In systems theory, all systems can be viewed as a part of a system, composed of systems and interacting with other systems (see fig.1.8). System's theory focuses on internal and external behavior of a system. It takes an interest in subsystems arrangement just as their relationships. Quite simply, we can say that systems theory studies the interactions between the system and its environment which is another system. A system is an organized structure or a collection of systems; put together, they present functionality which is absent when they are separate. We can classify the included subsystems into three systems (see fig.1.9):

- **Sensor system** :detects data coming from the environment so-called **input**.
- **Transformer system**: transforms the input perceived by the sensor system into output. Such transformation is so-called **throughput**.
- **Deliverer system**: delivers **output** to the environment.

We can find three kinds of classifications according to the systems:

1. Open, closed and isolated systems,
2. Reactive and cognitive systems,
3. Static and dynamic systems.

According to Bartalanffy in [11] an open system is a system that can't survive without continuous interaction and exchange with its environment. Such interaction is performed via two components: input and output. Hence, an open system is open to the environment and finds its behavior affected directly by the outside changing. In general, a system has many types of input such as energy, information or organization.

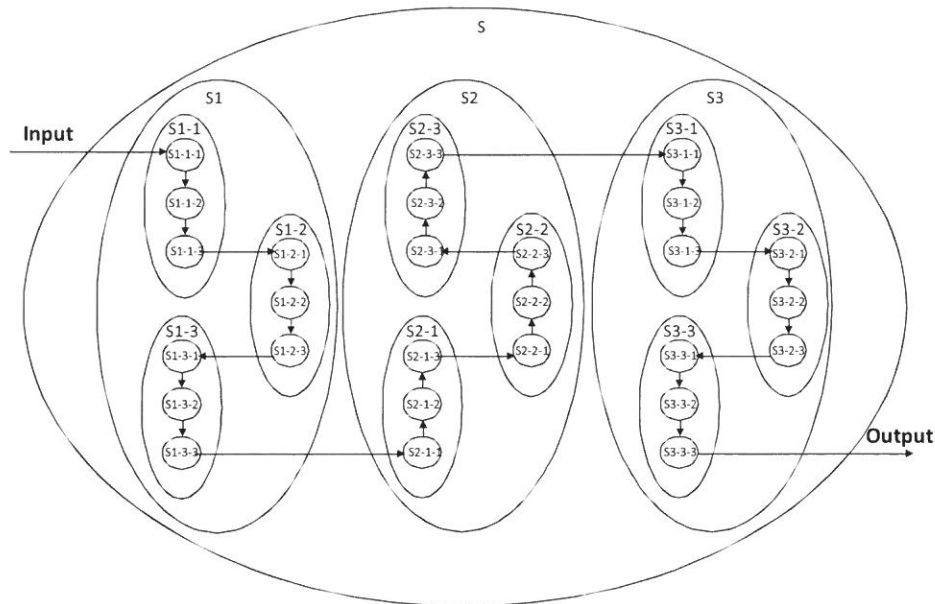


Figure 1.8: Hierarchy and interaction between systems

A system is said to be closed when it doesn't let one or more input pass through the system. For example, systems closed to energy are referred to **autark** systems, systems closed to information are called **independent** systems, and systems closed to organization are called **autonomous** systems [11].

An isolated system has no interaction at all with the environment. Hence, during the process, no data are exchanged with the external environment and the variations remain purely internal. This kind of system is uncontrollable [11].

On the other hand, a reactive system acts immediately in response to the input and uses predefined mechanisms [11].

A cognitive system can be thought as a system able to react through its reasoning mechanism, such as knowledge base, believe functions. . . [11].

A system is called static when it finds its behavior affected only by the input data entered from the environment [11].

A dynamic system, in contrast, finds its behavior changing not only as a result of the actions taken by the environment but also by an internal changing [11].

1.2.2 Agent based system

As we said above, this research aims to resolve the problem of interactions between variables in MTS by using artificial intelligence. We thought so to use intelligent agents. Each agent will deal with one physiological variable, it incorporates univariate time series technology to evaluate and forecast

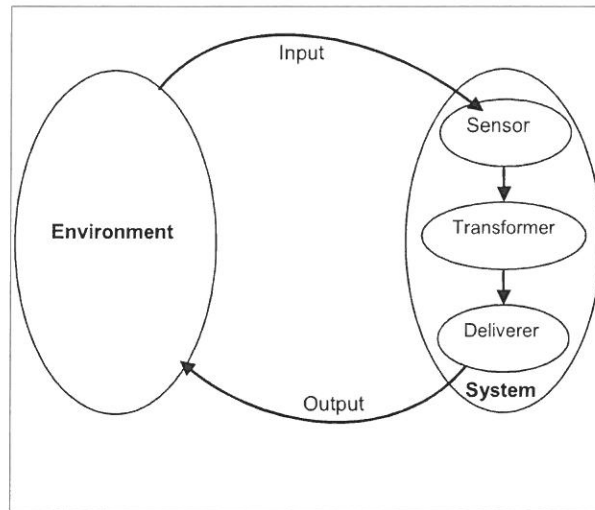


Figure 1.9: Systems interact with the external environment through Sensors and Deliverers

subsequent values in order to detect time series feature or anomalies.

With regard to system's theory that we introduced above, our system will be composed of three subsystems :

1. the sensor that we call the acquisition system,
2. the transformer that we call the monitoring system,
3. and the deliverer called the alarm system.

Each component or system is based around software agents. The acquisition agents are deployed across the acquisition system, a set of control agents and graphical agents are deployed on the monitoring system and the alarm agent is deployed on the alarm system. Each control agent is responsible for monitoring the behavior of each physiological variable.

1.2.3 Signal processing and forecasting

The research uses a statistical model which allows for evaluating the variable behavior in a fast and highly adoptable system. The first aim of this research is to forecast future patient state in order to take action at once. So the most important for us is to use an efficient forecasting model. A big number of forecasting methods exists in the literature and the number of studies examining the forecasting performance of each method is appreciably huge. Hence, a natural question to ask in this thesis is which forecasting method is best overall? The decision was based on a survey of the forecasting competitions between linear and nonlinear models held in the literature. Several papers devoted to comparing forecasting methods. Zhang et al. in [88] provide a survey of comparing time series forecasts from Neural

Network (NN) and linear models. Stock and Watson in [75] conducted a forecasting comparison between an impressive number of models. The contribution of this study included the use of 215 time series data at three forecasting horizons over the period 1959-1996, the use of 121 linear and nonlinear models which are divided into two different sets of forecasts: the linear models (AR and smooth exponential (EX)); the nonlinear models (STAR and NN), the investigation of unit root pretested models, and an extensive investigation of forecasting pooling procedures. They concluded that the nonlinear forecasting models fared relatively poorly for the time series they considered. In addition they prove that simple linear AR model made the lesser degree of forecast errors. Tkacz in [80], Marcellino in [56], Rech in [73] and Heravi et al. in [41] examine time series forecasting with linear and NN models. The general conclusion from the papers cited above appears to be that, there is non significant gain from using NN models for forecasting instead of the simple linear AR model. Teräsvirta et al. in [79] re-examined these issues. They used 47 nonlinear time series. Their results show that STAR models forecast considerably and consistently more than AR models and conclude that it is essential to first test linearity of time series before considering any model and conclude that the data used in [75] contained a linear components, that is, they fit a linear AR model. Thus, we can conclude that AR models come at the top of the linear models list and STAR at the top of nonlinear one.

In literature dealing with medical time series, AR models were used in their extended and general form $ARIMA(p, d, q)$, [49], [45], [65], [?], d can be different to zero in the case of non-stationarity [78]. In this thesis we will use, hence, the $ARIMA(p, d, q)x(P, D, Q)_s$ form in concern to have non stationary and seasonal time series under control.

1.3 Some definitions

In this section we will clarify some concepts. Often authors use one of the terms as methodology and method or technology and technique as though they were synonymous. But they aren't. So, in order to correctly use them, we shall start by proposing a distinction between them. Agazzi in [1] thought that the suffix "ology" that we find in both terms methodology and technology, gives a theoretical dimension or a scientific aspect that is usually bound up with it's use. In fact, the term methodology is already included this theoretical aspect as it's defined in some dictionaries. It's defined in WordNet 2.1 as : "*the branch of philosophy that analyzes the principles and procedures of inquiry in a particular discipline*", or "*the system of methods followed in a particular discipline*" and in Merriam-Webster's Online Dictionary, as : "*a body of methods, rules, and postulates employed by a discipline : a particular procedure or set of procedures*" . The Free Dictionary, defines methodology as "*the theoretical analysis of the methods*" or as "*the body of methods and principles particular to a branch of knowledge*". It adds that, in recent years, "*methodology has been increasingly used as a pretentious substitute for method in scientific and technical contexts*", and replies that "*the misuse of methodology obscures an important conceptual distinction between the tools of scientific investigation (properly methods) and the principles that determine how such tools are deployed and interpreted*". In an other section, it defines a method as

" means or manner of procedure, especially a regular and systematic way of accomplishing something". A method is also defined in Merriam-Webster's Online Dictionary as "a systematic procedure, technique, or mode of inquiry employed by or proper to a particular discipline or art". A technique is defined in WordNet 2.1 as "a practical method or art applied to some particular task" and in [1] as the "knowledge of doing or making". Agazzi in [1] defines also a technology as a "science that furnishes the theoretical grounds for officious doing" and adds that technology is "not only rapidly led to detailed knowledge of the natural world, which allowed for more adequate explanations of the success of many techniques already used; it also inaugurated a process by which newly acquired knowledge was immediately applied toward the creation of new techniques and was even sought for the sake of some technical application". In encyclopaedia Britannica technology is "the application of scientific knowledge to the practical aims of human life or, as it is sometimes phrased, to the change and manipulation of the human environment. Technology includes the use of materials, tools, techniques, and sources of power to make life easier or more pleasant and work more productive. Whereas science is concerned with how and why things happen, technology focuses on making things happen. Technology began to influence human endeavour as soon as people began using tools. It accelerated with the Industrial Revolution and the substitution of machines for animal and human labour. Accelerated technological development has also had costs, in terms of air and water pollution and other undesirable environmental effects".

We found also that authors use interchangeably the two terms prediction and forecasting. In wordNet 2.1 The term forecasting has one sense : "prediction, foretelling, forecasting, prognostication – (a statement made about the future)" and the verb forecast has three senses : "1. forecast, calculate – (predict in advance), 2. calculate, estimate, reckon, count on, figure, forecast – (judge to be probable), 3. bode, portend, auspicate, prognosticate, omen, presage, betoken, foreshadow, augur, foretell, prefigure, forecast, predict – (indicate by signs; "These signs bode bad news")". So prediction or forecasting are two terms discussing the same statistical concept. So it was agreed that we use the term forecasting all along the thesis.

1.4 Learning outcomes of the thesis

The four main learning outcomes as defined by the Scottish Qualifications Authority (SQA) and presented in [15] for the thesis are :

LO1 "The creation and interpretation of new knowledge, through original research, or other advanced scholarship, of a quality to satisfy peer review, extend the forefront of the discipline, and merit publication".

The research work has defined a new agent-based system, based on an application of monitoring and forecasting method. It has merited publication at several key points:

1. Nouira K. and Trabelsi A., (2010), Online Multiagent Monitoring in Intensive Care Units, to appear in proceedings of International Colloquium on Computing, Communication, Control, and Man-

agement(CCCM 2010), IEEE Catalog Number: CFP1048E-PRT; ISBN: 978-4244-7380-9, August 20-22, 2010, Yangzhou China.

2. Nouira K., Ferchichi A. and Trabelsi A., (2007), Temporal Multi-Agent Based Architecture for Medical Monitoring Systems in Intensive Care Units: a Unified Systemic Approach, In proceedings of the Septièmes Journées Scientifiques des Jeunes Chercheurs en Génie Electrique et Informatique (GEI'2007), CPU, pp 109-118.
3. Nouira K. and Trabelsi A., (2007), Time Series technology in pattern recognition: case study in medical support systems, In proceedings of the 4th International Multi-Conference on Systems, Signals & Devices (SSD-07). IEEE. Conference on Communication and Signal Processing(CSP), volume 3.
4. Nouira K., Ferchichi A. and Trabelsi A., (2007), Temporal multi-agent system in intensive care monitoring, In proceedings of the 24th Annual Conference and Exhibition HEALTHCARE COMPUTING (HC2007). British Computer Society (BCS), pp. 38-46.
5. Nouira K. and Trabelsi A., (2006), Temporal Multi-Agent System in Intensive care Monitoring, Poster in International Symposium on Computational Biology & Bioinformatics, Bhubaneshwar, India.
6. Nouira K., Ferchichi A. and Trabelsi A., (2006), Architecture of Temporal Online Monitoring Multi-Agent System in Intensive Care Units, In the 7th Tunisia-Japan Symposium on Society, Science & Technology (TJSSST 2006), Sousse, Tunisia.
7. Nouira K. and Trabelsi A., (2006), Time Series Analysis and Outlier Detection in Intensive Care Data, In proceedings of the 8th International Conference on Signal Processing (ICSP'06). IEEE, 2006, ISBN: 0-7803-9736-3, vol 4, pp 2499-2502.

LO2 "*A systematic acquisition and understanding of a substantial body of knowledge, which is at the forefront of an academic discipline or area of professional practice*".

The work has involved investigating actual medical monitoring systems, and methods used to improve them. From this, an agent-based model was created for the new method. These experiments used a large amount of physiological variables and processed them to prove that the system can handle all kinds of information loads.

LO3 "*The general ability to conceptualize, design and implement a project for the generation of new knowledge, applications or understandings at the forefront of the discipline, and to adjust the project design in the light of unforeseen problems*".

The project idea started from the observation of how stressful and risky, an ICU when the actual monitoring systems decide to sound a huge number of alarms of which the majority are insignificant. The research also involved modelling typical physiological variables behavior using time series

technology. The research also used software agent technology that is most suitable for multivariate time series. Software agents are robust and highly adoptable software entities and, if used correctly, have minimum impact on the use of computational power, and memory usage of the system.

LO4 "*A detailed understanding of applicable techniques for research and advanced academic inquiry*".

In this research project, many new ideas and technologies were used from different disciplines : Intelligent agent technology, Time series technology and monitoring technologies in the intensive care units.

1.5 Structure of this thesis

The format of the thesis is as follows:

- Chapter 1: **Introduction**. This provides an introduction of the thesis. It defines the motivation for the work, and how well the research has matched the learning outcomes.
- Chapter 2: **Intensive Care Units Monitoring Systems Issues**. This outlines the different weak points of current monitoring systems in ICUs. It will highlight some important and dangerous aspects of the monitoring systems risks.
- Chapter 3: **Time Series Analysis in Medical Data**. This deals with Time Series Analysis, and gives a brief overview of statistical methods used to deal with Time Series Features. A particular focus will be on explaining anomalies detection methods, as these are used in the proposed model.
- Chapter 4: **Online Multi-Agent Monitoring System for Intensive Care Units**. This explains, in detail, the system architecture, the different parts running.
- Chapter 5: **Experimental Study**. This outlines the different components' development and the results of the experimentations as compared with results from other systems.
- Chapter 6: **Conclusions and future research**. This is the final chapter and deals with the main conclusions from the research, how successful it was, and proposes future research plans.

Chapter 2

MONITORING ISSUES IN INTENSIVE CARE UNITS

2.1 Introduction

An Intensive Care Unit (ICU) is a demanding environment due to the critical condition of patients and the variety of equipment necessary to monitor and support patients. ICU equipment may include : 1) patient monitoring equipment, 2) life support and therapeutic equipment, 3) diagnostic equipment.

- **Patient monitoring equipment.** Include the following:
 - *Physiologic monitor.* Can be configured to continuously measure and display a number of vital signs or physiological variables via electrodes and sensors that are connected to the patient. These may include the Electro Cardiogram (ECG), Respiration Frequency, Blood pressure and Blood Temperature. Physiologic monitors generate alarms in any change of patient conditions.
 - *Pulse oximeter.* Monitors the arterial hemoglobin oxygen saturation (SpO₂) of the patient's blood with a sensor clipped over the finger.
 - *Intracranial pressure monitor.* Measures the pressure of fluid in the brain.
 - *Apnea monitor.* Continuously monitors breathing via electrodes or sensors placed on the patient's thorax. It displays respiration parameters, and generates an alarm if a certain amount of time passes without a patient's breath being detected.
- **Life support and therapeutic equipment.** Intensive care equipment for life support include the following:
 - *Ventilator.* Also called respirator, assists pulmonary ventilation in patients who cannot breath on their own. It's microprocessor-controlled and programmable. It regulates the volume,

pressure, and flow of patient respiration. Ventilator have also a physiologic monitoring functionality and generates alarms in case of therapeutic malfunction.

- *Infusion pump.* Infusion pumps employ automatic, programmable pumping mechanisms to deliver continuous anaesthesia, drugs, and blood infusions to the patient. It can also generate alarms in case of therapeutic malfunction.
- *Crash cart.* Emergency equipment includes a defibrillator, airway intubation devices, a resuscitation bag or mask, and medication box. Crash carts are strategically located in the ICU for immediate availability in case of cardiorespiratory failure.
- *Intra-aortic balloon pump.* A device that increases cardiac output and coronary blood flow for patients with unstable angina, myocardial infarction (heart attack), or patients awaiting organ transplants. It has a console which displays heart rate, pressure, and electrocardiogram (ECG) readings.

- **Diagnostic equipment.** Intensive care equipment for diagnostic include the following :

- Mobile X-ray units are used for bedside radiography on patients who cannot easily get to an X-ray room.
- Portable clinical laboratory equipment, are used for blood analysis at the bedside.

Both monitoring and therapeutic equipment have alarm capabilities and generate optic and acoustic alarms each time when a change of patient's condition or a malfunction of the equipment occurs.

Most of the time, alarms from therapeutic equipment are technically correct because they result from technical measurements inside the device. But they may not always match the criticability of the clinical situation despite the correct measure. On the other hand, alarms from monitoring equipment can be also technically incorrect in that the signal acquisition may be distorted [47].

Unfortunately, a lot of defects are present in ICU and medical errors occur with alarming frequency. There is a strong evidence that practically every patient admitted to an ICU will experience a potentially life-threatening medical error [26]. Such errors, according to Donchin and Seagull in [27], are due to the difficult working conditions in the ICU. In fact the high volume of patient data, the ambient noise, the lighting level and the scheduling were all identified as likely factors contributing to the errors in ICUs.

- **High volume of patient data.** In ICUs the staff is confronted to a flood of recorded variables relevant for patient monitoring such as physiological variables, laboratory data, device parameters, etc. This high dimensional data exceeds the human ability to identify and develop a solution considering all the observations [32].
- **Ambient noise.** In today's ICUs environment, monitoring systems revolve around information that is recorded by electronic patient monitoring devices. Such devices have steadily advanced in

recent years. In fact significant improvements have been made in their ability of measurements [17]. But they still typically incorporate a basic algorithm using a standard threshold filter which sounds an alarm as soon as a monitored value (single signal) goes above the set upper limit or below the set lower limit [74] defined beforehand by the medical staff (see fig.2.1). Hence, depending on the boundary-based alarm [74], current monitoring systems are perceived as unhelpful by medical staff because of the huge number of alarms which are result of erroneous readings or with no clinical significance [17], [74]. Indeed, we can obviously say that it is not the best method of event detection. Because, in many cases, it is not the alarm itself but rather the clustering of repetitive alarms from multiple devices that attracts attention [74] and the information that a physician needs, is not the crossing a threshold by a single signal, but it is the detection of relevant abnormalities and changes which are reflected by the analysis of the evolution of a combination of some physiological parameters working together[17]. In other hand, we can say that available monitoring systems can monitor more than 40 physiological variables [17], [89] and automatically set an alarm for each one. In addition to that, the present technology consists of sounding audible alarm and each alarm produces noise louder than 80 *dB* [17]. Moreover, whatever degree of the emergency, the monitoring system produces the same audible alarm [82].

- **Long hours schedule.** According to the Association of Professional Sleep Society, the night-time operator's fatigue contributed to the disasters such as Chernobyl and three-mile island. This is due to the fact that a single night of continuous sleep deprivation causes deterioration in performance similar to that induced by a blood alcohol level of 0.1% [19], [52]. Hence, the extended hours typically worked by nurses, interns and residents, when combined with the challenging working conditions of the ICU, are likely to produce medical errors and that the consequence are apt to be more serious, given the vulnerability of ICU patients [19].

According to Donchin et al. in [26], human errors in ICUs cause 29% of patient status deterioration and 12% of deaths. Because those defects are common, they feel normal and inevitable. Instead of trying to fix them, people accept them. Many nurses, doctors, patients and families are all too familiar with what went wrong in ICU and think that completely preventing errors is a hopeless task [12]. [30] proved that errors are not only human but in 17.2% were due to technical equipment failure. Equipment failure in ICU represents a very common source of serious adverse incidents. This is due to the fact that 1) there is no regular checking and maintenance even for old and new equipment [71]. 2) There is an incomplete staff training and education before using of new equipment inserted into ICU [71]. 3) And probably a manufactured dysfunctioning [59].

This chapter provides an understanding of ICU issues and the efforts used, in the first hand, to improve the quality, safety, efficacy, and cost of health care, and in the second hand, to promote information technology. It outlines various weaknesses present in current ICUs and how they reverberate on the patient state and ICU staff working conditions.

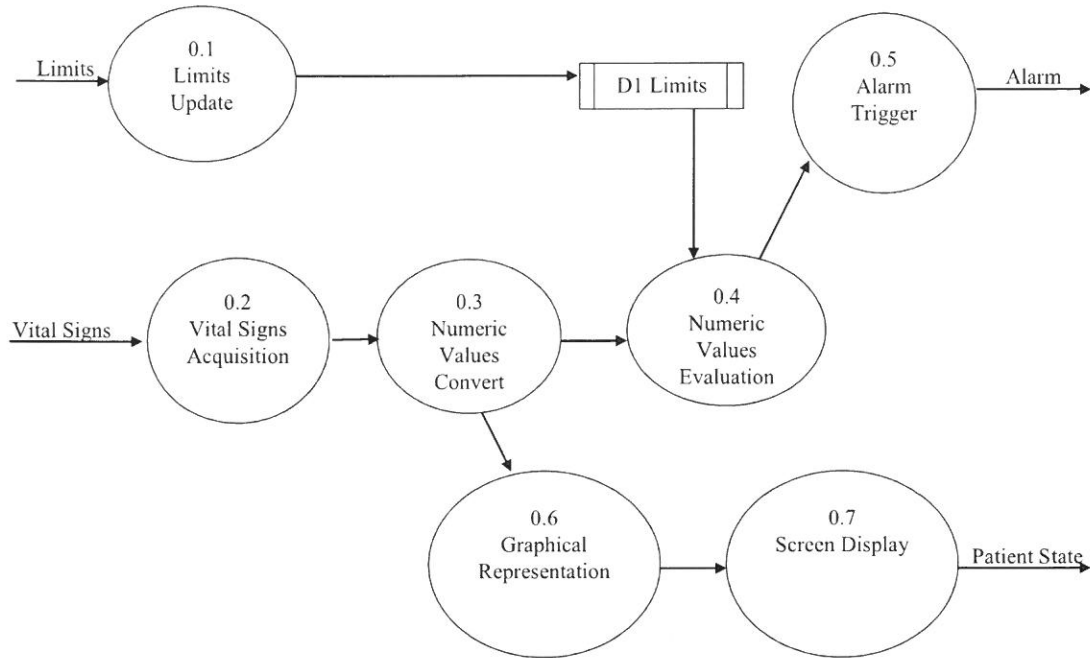


Figure 2.1: Data Flow Diagram of Alarm System in Intensive Care Monitoring System

Our contribution is focused on improving ICU work conditions and more exactly reducing noise produced by patient monitoring equipment alarms. For this, many different techniques are employed to tackle the same point of failure: filtering insignificant alarms. Our proposal deals with the condition when patient monitoring equipment record data, but start to misbehave either to read wrongly or to trigger insignificant alarms.

2.2 Patient monitoring systems issues

As Hudson "*Monitoring is making repeated or continuous observations or measurements of the patient, his or her physiological function and the function of life support equipment, for the purpose of guiding management decisions, including when to make interventions and assessment of those interventions*" [42, p 630].

Monitoring systems can loosely be described as a set of high-technology patient monitoring devices whose job is to report every physiologic signal that may possibly be measured. Such systems are present in ICUs, where we need to monitor the patient current state. Technological advances, in this field, hold great potential in improving patient care but in an other hand the current stage of algorithm development represents a big trouble in the ICU. In fact, alarm soundings in the ICU are frequent but unfortunately insignificant more commonly than significant [82]. Tsien in [83] asks about 86% of all alarm

Category	Significance	Exemple
True Positive, Clinically Relevant (TPR) Alarm	Alarm was appropriate given the actual data value as compared to the set threshold value, and the patient's condition required attention	Hart rate = 200 beat per minut (bpm), monitor set with an upper threshold at 160 bmp, and the monitor sounds an alarm
True Positive, Clinically Irrelevant (TPI) Alarm	Alarm was appropriate given the actual data value as compared to the set threshold value, but the patients condition had not changed in a way that requires additional medical attention	Patient's systolic blood pressure transiently crosses the set upper threshold during endotracheal suctioning
False Positive (FP) Alarm	Alarm was inappropriate given the actual data value	Hart rate = 80 bpm, and the monitor sounds an alarm
False Negative (FN) Alarm	Acute patient condition occurred but no alarm was triggered	Hart rate = 200 bpm, and no alarm is generated
True Negative (TN) Alarm	Appropriate monitor silence when no alarms were expected	Hart rate = 80 bpm, and no alarm is generated

Table 2.1: Tsien classification of alarms [78]

soundings are insignificant and Schoenberg et al. in [74] said that, in some cases, over 90% of all alarms produced by most devices are false. Here, the term false alarm is used to point out both technically false alarms as well as alarms based on true readings but are considered to be clinically insignificant. Tsien in [83] classified the recorded alarms into one of the following categories (see table2.1).

In a separate classification, all alarms were categorized where possible as either "significant alarms" or "insignificant alarms". Significant alarms refer to those alarms clearly associated with an intervention of the ICU staff. Insignificant alarms, on the other hand, refer to those alarms clearly not associated with the staff intervention [17], [74].

In the following we present some monitoring algorithms proposed in the literature trying to decrease the number of insignificant alarms.

2.3 Attempts to reduce false alarms using statistics

Existing monitors, as mentioned earlier, have exorbitant false alarm rates. And when nurses and physicians are frustrated by the flood of the noise, they will try to find their own solution either by muting notorious devices or setting alarm limits that are unlikely to be exceeded [74]. As Schoenberg et al. in [74], no studies has qualified the effects of this action on patients but it is evident that events are missed and therefore decision can be erroneous. So it is obvious that the development of better and smarter algorithm is still needed. The aim of this algorithm is to point out those alarms designed as

meaningful by the ICU staff and filter out all others. Such attempts to decrease the number of false alarms in ICUs seem worthwhile. But are those filters reducing the insignificant alarms without missing the significant ones? So, efforts to Improve alarm specificity, however, must not overlook the need to maintain alarm sensitivity [82]. And an ideal algorithm is that both sensitivity and specificity of the alarm system are 100%. Sensitivity means that all significant alarms are triggered and specificity means that all alarms triggered are significant. Some attempts, as presented in [81], are based on filters such as moving average filter, moving median filter, delay filter and Sampling rate filter.

- **Moving average filter based algorithm.** The moving average filter based algorithm takes as input a window size of value N and calculates the relative average which is compared against the threshold limits. Once the value $N + 1$ have been read, the window is translated and the algorithm will compute the newly average. If the average value is outside the limits, the alarm will be generated.
- **Moving median filter based algorithm.** The moving median filter based algorithm is working identically as the moving average, except that in this time the median value is used for checking against threshold limits rather than the average.
- **Delay filter based algorithm.** The delay filter based algorithm takes as parameter an integer N and sounds an alarm only when a sequence of N values are outside the limits.
- **Sampling rate filter based algorithm.** The sampling rate filter based algorithm takes also as parameter an integer N and only one value in each N values will be tested against the limits.

In considering the results presented in [81], those filters eliminate significant alarms to a greater extent than false and insignificant alarms. In other hand, the experimentations were done on data with a big number of alarms and when there are very few alarms for a particular physiologic signal type, significance of the results is difficult to ascertain. An other significant drawback of such filters is the fact that it works on a single signal. So nothing can prove if they will have the same effectiveness when used on data in conjunction with other signals.

2.4 Attempts to reduce false alarms using artificial intelligence

In medicine, symptoms are usually based on a combination of many physiological signals' trends. Hence, when we try to filter alarms in ICUs, the more efficient filter may be the multi-signal one. Artificial intelligence has offered a lot of algorithms in this field. But, until few years ago, the monitoring devices in ICUs were discrete and not interconnected. Hence, the employment of such meaningful algorithms based on multi-signal control was impossible. This is due to the fact that those algorithms require that data be saved over time in a central data base. It was not until recently that PC-based computer systems targeted these goals and allowed real-time data acquisition from multiple devices are brought into the

	Traditional System	Trend-based system
Number of alarms generated	6872	544
number of insignificant alarms	6575	302
Number of significant alarms	297	242
Sensitivity	100%	81.4%
Specificity	4.2%	44.4%

Table 2.2: Comparison between single parameter boundary-based module and trend-based module

ICUs [74]. Hence, the availability of data in a programmable environment made implementation of all those algorithms developed before this area in artificial intelligence possible.

In this section we will present a trend-based algorithm presented in [74]. This algorithm was used to develop a computerized module that implemented as part of iMD soft's MetaVision paperless ICU suite [43]. Such system acquires the data directly from monitors, ventilators and infusion pumps and records it into a central SQL data base. The aim of this algorithm is to identify the physiological trends that are determinantal to the patient. Each trend, even when it crosses the limit, will not be considered, as long as there is no other physiologic trends to show a specific state. Hence, alarms are based on the identification of a combination of trends and not a singular one. Such algorithm reduces significantly the false alarms due to erroneous readings because they are unlikely to produce a trend and even it will be produced, it will be without correlation with other variables. Instead of comparing each value with a threshold, this algorithm uses the physiometry trend, defined by the user, for each signal. For example, the change of the systolic blood pressure between the current minute and five minutes ago. Then a criterion will be assigned for each physiologic trend (e.g., systolic blood pressure > 160 *mgh*). Each criterion can be true, false or unknown (if missing value). After that a score is affected for each criterion value. And so, the sum of a predefined combination of scores at any point in time is compared with a threshold value to determine whether or not an alarm should be activated.

The performance characteristics of this module are compared to traditional single parameter boundary-based module and results are resumed in table 2.2.

Considering that the yield of those alarms depends on their sensitivity and specificity, we can say that the algorithm has reduced the insignificant alarms (so increased the specificity) but it is not accurate enough to point all significant ones. In fact, the module's sensitivity was 81.4% and it is failing to detect 18.6% of significant alarms as defined by the ICU staff. Hence, as Schoenberg et al. in [74], it's obvious that "reaching a consensus on the right logic to identify a significant alarm and the definition of significance is an inacheivable goal".

2.5 Attempts to reduce false alarms using standardization

As seen above, available monitoring systems provide a great number of potential audible alarms. Thus, both major event, that was related to worsening for the patient status, and minor ones, as alarms

Priority	Description	Exemple
High	Urgent situation which leads immediatly to a vital problem and requires an immediate attention from the medical staff	Electric or pneumatic failure, Paw high, continuous pressure
Medium	Potentially serious situation and needs prompt attention	SpO2high or low, perfusion pump empty, nutrition pump empty
Low	Minor condition and so only an attention is needed	Sensor failure, perfusion pump low, nutrition pump low

Table 2.3: Significance of alarm level priority

generated by the perfusion or the nutrition pump, sound the same. Such situation can be confusing for medical personnel trying to determine the source of the alarm. This is due to the fact that there is no standard for alarm setting [17]. Here, the use of priority in alarm management is proposed to recognize the kind of the alarm, if it is major and needs an immediate intervention from the staff or if it is a minor one. The concept of priority was treated by several committees for normalization that define standards for medical devices [17]. These standards provide on the first side, a classification of the alarms according to a level of emergency (high, medium and low) with audible characteristics corresponding to each of these levels see table2.3. On the other side, for each monitoring system, the events or parameters that should provide an audible alarm with a given degree of emergency [17].

Very few monitoring systems currently use these standards, we can hold up as an example a prototype of a system for patient monitoring in ICU called MetaMonitor developed in Carnegie Mellon University. This system, even if it doesn't attempt to reduce the number of alarms, it reduces the number of alarm sounds by providing easily accessible information. In fact, in current systems, data is displayed into small monitors hanging on the wall. So, because it's difficult to see from a distance such system requires audible alarm triggering.

But in MetaMonitoring system, see fig.2.2, we can see networked displays which can access any information anywhere in the unit, in addition to that, lights outside each room flash different colors for different types of alarms and single speakers replacing audible alarm on each monitoring device. We see, also, wearable armband devices providing individual alarm notification for each nurse and contains speaker, vibrating motor, LCD touch screen, etc...

Metamonitor knows which nurse is responsible for which patient. Different Alarm notifications are given to each via the armband device. It knows also where nurses are within the unit by using a real time location system. This information is used to give different notifications when a nurse is already in the patients room, to automatically silence alarms when nurses enter rooms, and to show relevant information on the networked display nearest to nurses. By reserving general audio alarms for emergencies, metamonitor cuts down on the noise in the ICUs, easing stress on patients.



Figure 2.2: MetaMonitor devices. Source [29]

The review of the current monitoring system permits the conclusion that the alarm generation is still the source of a big number of false alarms only considered as low or may be medium priority alarms.

2.6 Conclusion

With all of these shortcomings of current monitoring systems in ICU and the attempts to reduce insignificant alarms, what are some possible directions for improving the situation? Several approaches to intelligent monitoring techniques seen above look appealing. But, which of these approaches is likely to be most promising? It is difficult to say. What is not as clear is how to go about effectively decreasing the insignificant alarm rate without missing significant ones. In other terms, how can we increase the specificity of those systems without decreasing their sensitivity.

In this work we try to implement an alternative approach to the problem. Our exploration will especially be the combinations of statistical technology as well as of multi-agent one. This work seems imperative for continued progress in the area of ICU patient monitoring.

Chapter 3

TIME SERIES ANALYSIS IN MEDICAL DATA

Principal results of this chapter are published in the 8th International Conference on Signal Processing (ICSP'06) [61].

3.1 Introduction

Our special focus is the use of time series technology in intensive care monitoring not only to detect critical states but also to forecast future states in order to take the efficient decision. Time series analysis allows to detect characteristic features such as outliers, level changes, trends and changepoints which are important for assessing critical states [32]. Usually the detection of a changepoint which means the occurrence of a level change, a trend, a variance change or any other characteristic features change [21] in a system leads to an alert, which allows the physician to check the situation and possibly take some therapeutic actions. A well known problem here is the occurrence of isolated and patchy outliers which make time series noisy. They can produce false alerts and can mask changepoints (see fig.3.1). Moreover, *"outliers may be of fundamental interest in themselves and therefore their identification should also be considered as a goal in itself"* [9, p947]. *"Hence there is a necessity for an automatic detection of outliers in on-line monitoring data"* [5, p2].

There are various statistical approaches such as nonlinear time series analysis, neural networks frequency domain analysis [5] and genetic algorithms [3] which are dealing with time series. Outliers have been already treated by using ARIMA models [45], state space models [4], [5], dynamic models [76], graphical models [39], Gibbs sampling [49] and some robust estimators [10], [9], [49].

Several procedures are available in the literature to handle changepoints in time series. A prominent method for the detection of changepoints is the multiprocess kalman filter [23], [22], [76] we can find also some stochastic models as described in [21].

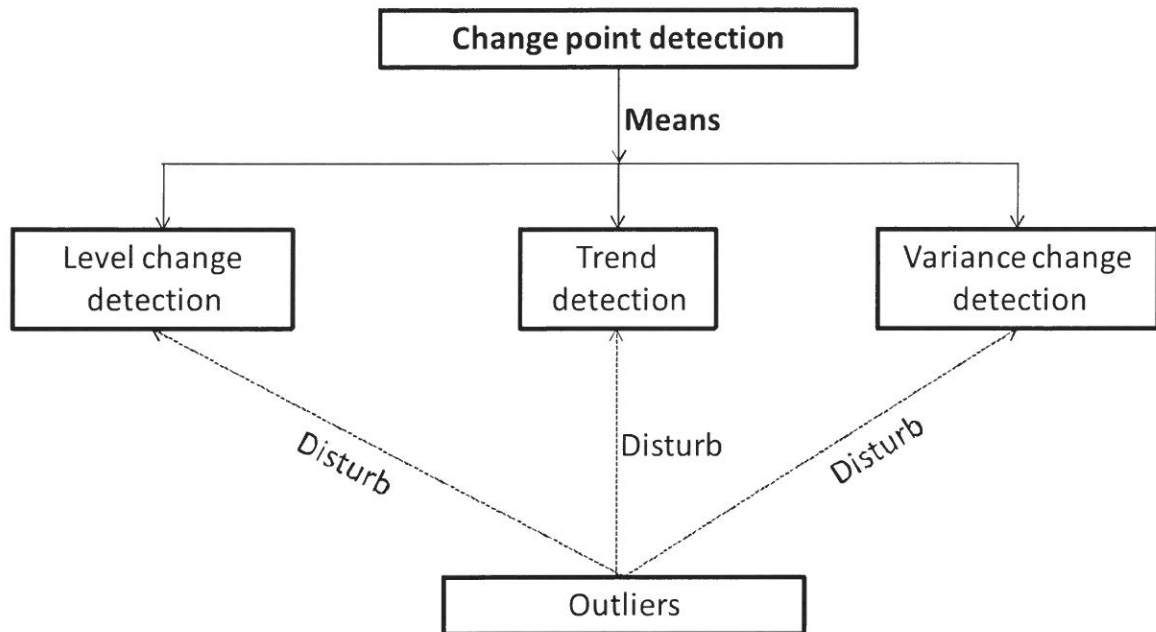


Figure 3.1: Outliers give false alert and mask changepoints

Some authors dealt with level change detection by using analytical methods based on phase space reconstruction [4] and simple median filtering [84].

Some studies were concerned with automated model detection in on-line monitoring data [32], [39], [78], [70], [51], [69], [86], [34], [37], [45], [72] in order to forecast the patient state and avoid disasters.

Because intensive care on-line monitoring data is generally high dimensional and multivariate, a lot of literature [7], [38], [35], [16] and [87] dealt with the dimension reduction which reduces the dimension of the vector by taking the most important data without loss of information [87] in order to be able to take decision.

We find an increasing amount of literature in this field and there seem to be two major approaches :

- Patient data can be seen as a set of univariate time series and we analyze each series alone.
- Patient data must be seen as a multivariate time series and we must find the adequate framework for analyzing this series.

This paper is organized as follows. In the following section we present some works dealt with outlier detection. Section 3 contains works dealt with level change detection. Section 4 deals with changepoint detection. Section 5 presents some model recognition in on-line monitoring time series and finally in section 6 we will try to present solutions to high dimensionality and multivariate time series in intensive care on-line monitoring, some conclusions are given in section 7.

3.2 Outlier detection

Data from intensive care patient monitoring may be noisy and exhibits outliers. Hence, automated detection of outliers will play a prominent role in on-line monitoring. Therefore, we see a lot of literature dealing with outlier detection namely [4] which proposed a new graphical approach for model recognition in univariate time series from on-line monitoring systems in intensive care. This approach is based on phase space reconstruction which allows time series to be a multivariate sample.

Let $\{y_t\}_{t \in \{1, \dots, N\}}$ be a time series, the m -dimensional vectors which are the time delayed observations are defined as follows :

$$\vec{Y}_t := (y_{t+(m-1)T}, \dots, y_{t+2T}, y_{t+T}, y_t)' \in \mathbb{R}^m \quad (3.1)$$

with T and $m \in \mathbb{N} \setminus \{0\}$, and $t = 1, \dots, N - (m - 1)T$. T is called the time delay and must be chosen such that $\text{Corr}(Y_t, Y_{t+\tau}) = 0 \forall \tau \geq T$ [5]. m is called the embedded dimension and recommended to be chosen such that $m = 1 + \max_{\lambda} \{\lambda; |\Phi(\lambda)| \neq 0\}$ and $\Phi(\lambda)$ is the Partial Autocorrelation Function (PACF) [5]. So that the univariate Time series is transformed into an m -dimensional space and the set

$$\{\vec{Y}_t / t = 1, \dots, N - (m - 1)T\} \quad (3.2)$$

forms the phase space reconstruction. This concept presents a potential in outlier detection. In fact, if an outlier arises in the phase space vectors \vec{Y}_t and \vec{Y}_{t+1} these vectors extrude from the regular observations and cause typical movement through the phase space.

Later, [5] demonstrate that neither state space models nor ARIMA-models are robust against multiple or patchy outliers. They said that if outliers arise within less than ω time interval distance of each other those models can fail to detect them. Hence they constructed an automatic procedure so-called α -outlier concept for the detection of outliers in on-line monitoring time series. This concept leads to a multivariate sample with dependent observations. So they used the phase space reconstruction.

The main idea of this concept is that α -outliers may have an extreme position in a region so-called α -outlier region defined by :

$$\text{out}(\alpha, \mu, \Sigma) := \{x \in \mathbb{R}^m; (x - \mu)' \Sigma^{-1} (x - \mu) > \chi_{m, 1-\alpha}^2\} \quad (3.3)$$

with respect to $N(\mu, \Sigma)$.

For a sample of N elements we can speak of α_N -outlier region which must respect the following condition :

$$P \left(\bigcap_{i=1}^N X_i \in \mathbb{R}^m \setminus \text{out}(\alpha_N, \mu, \Sigma) \right) = 1 - \alpha \quad (3.4)$$

with $X_i \sim N(\mu, \Sigma)$ ($i = 1, \dots, N$) and $\alpha \in (0, 1)$.

To identify all outliers in data we must evaluate α_N -outlier region. But this region depends on two unknown parameters such as μ and Σ . Therefore, we must estimate this region. In their paper

they discussed the Minimum Volume Ellipsoid (MVE) estimators which can be used to calculate robust estimators of the process parameters.

Becker and Gather investigated α -outliers concept in order to improve outlier identification rules. They used the masking breakdown point criterion in [9] and maximum asymptotic bias in [8] despite their failure to detect outliers when some extremely outlying observations happen. They thought that it's possible to improve this criterion by using some estimators.

Because outlier identification means outlier region evaluation as said above, and an outlier region depends on two unknown parameters μ and Σ that we must estimate by respectively m and S -estimators. Either [9] or [8] used high breakdown estimators. They found a good result in [9] where they used S -estimators. But in [8] they found that the use of MVE, MCD and S -estimators with maximum asymptotic bias to get an outlier identifier in presence of high breakdown points does not succeed.

[10] discussed the behavior of three outlier identifiers based on robust estimators such as Minimum Volume Ellipsoid (MVE), Minimum Covariance Determinant (MCD) and S -estimators based on beweight function (BW) such identifiers are so-called respectively OR_{MVE} , OR_{MCD} , OR_{BW} . By varying the sample size, results differ but they conclude that OR_{BW} should be favoured in the case of a sample of moderate dimension.

[55] used Gibbs sampling to identify outliers in autoregressive models. Let $\{y_t\}$ be a set of data points observed, say

$$y_t = \delta_t \beta_t + x_t \quad (3.5)$$

where $\{x_t\}$ is an autoregressive process of order p , such that

$$\Phi(B)x_t = \Phi_0 + a_t \quad (3.6)$$

with

$$\Phi(B) = 1 - \Phi_1 B^1 - \Phi_2 B^2 - \dots - \Phi_p B^p \quad (3.7)$$

Φ_0 is a constant

and

a_t is a white noise independently distributed $N(0, \sigma_a^2)$

δ_t is an outlier indicator it can take values 0 or 1

$$\begin{cases} \delta_t = 1 & \text{if the } t\text{-th observation is an outlier} \\ \delta_t = 0 & \text{otherwise} \end{cases} \quad (3.8)$$

we can consider $\delta = (\delta_1, \dots, \delta_n)'$ as an outlier indicators vector.

β_t denotes the magnitude of y_t if it is an outlier and we can consider $\beta = (\beta_{i+1}, \dots, \beta_n)'$ as an outlier magnitude vector.

Assume that the first outlier occurs at time $t = i$ so we can say that the outlier indicator vector becomes

$$\delta = (\delta_{i+1}, \dots, \delta_n)' \quad (3.9)$$

and the outlier magnitude vector becomes

$$\beta = (\beta_{i+1}, \dots, \beta_n)' \quad (3.10)$$

and the observed series becomes

$$y_t = x_t \quad t = 1, \dots, i \quad (3.11)$$

and

$$y_t = \delta_t \beta_t + x_t \quad t = i + 1, \dots, n. \quad (3.12)$$

- Prior distributions

assume that

$$\Phi = (\Phi_1, \dots, \Phi_p)' \quad (3.13)$$

and

$$p(\delta_t = 1) = \alpha \quad (3.14)$$

it seems that each time point y_t has the same prior probability α of being an outlier for $t = i+1, \dots, n$.

Hence, we can say that the prior distributions are :

$$\Phi \sim N(\Phi_0, \Omega^{-1}) \quad (3.15)$$

$$\delta_t \sim \text{Bernoulli}(\alpha); \quad t = p + 1, \dots, n \quad (3.16)$$

$$\beta_t \sim N(0, \tau^2) \quad (3.17)$$

$$\alpha \sim \text{Beta}(\gamma_1, \gamma_2) \quad (3.18)$$

$$\frac{v\lambda}{\sigma_a^2} \sim \chi_v^2 \quad (3.19)$$

assume that $\Phi_0, \Omega, \lambda, v, \gamma_1, \gamma_2$ are known.

- Posterior distributions

The conditional posterior distribution of the AR parameter vector Φ as given by

$$\Phi | y, \sigma_a^2, \delta, \beta \sim N(\Phi^*, \Omega_*^{-1}) \quad (3.20)$$

$$\text{where } \Omega_* = \frac{\sum x_{t-1} x'_{t-1}}{\sigma_a^2} + \Omega; \quad \Phi^* = \left(\frac{\sum x_{t-1} x'_{t-1}}{\sigma_a^2} + \Omega \right)^{-1} \left(\frac{\sum x_{t-1} x'_{t-1} \hat{\Phi}}{\sigma_a^2} + \Omega \Phi_0 \right).$$

The conditional posterior distribution of the innovational variance σ_a^2 is

$$\sigma_a^2 | y, \Phi, \delta, \beta \sim \text{inverted} - \chi^2. \quad (3.21)$$

The conditional posterior distribution of δ_j for $j = i + 1, \dots, n$

$$\delta_j \sim \text{Bernoulli}(\alpha) \quad (3.17)$$

with probability

$$p(\delta_j = 1 | y, \Phi, \sigma_a^2, \delta_{(j)}, \beta, \alpha) = \frac{\exp\left(-\frac{1}{2\sigma_a^2} \sum_{i=j}^{T_j} e_t(1)^2\right) \alpha}{\exp\left(-\frac{1}{2\sigma_a^2} \sum_{i=j}^{T_j} e_t(1)^2\right) \alpha + \exp\left(-\frac{1}{2\sigma_a^2} \sum_{i=j}^{T_j} e_t(0)^2\right) (1-\alpha)} \quad (3.18)$$

where $\delta_{(j)} = (\delta_{i+1}, \dots, \delta_{j-1}, \delta_{j+1}, \dots, \delta_n)'$

$T_j = \min(n, j + i)$ and $e_t(\delta_j) = x_t - \Phi_0 - \sum_{q=1}^i \Phi_q x_{t-q}$ is the residual at time t

The posterior distributions of the outlier magnitudes β_j for $j = i + 1, \dots, n$ are

$$\beta_j | y, \Phi, \delta, \beta_{(j)}, \sigma_a^2 \sim N(\delta_j \beta_j^*, \sigma_j^2) \quad (3.22)$$

where

$$\beta_{(j)} = (\beta_{i+1}, \dots, \beta_{j-1}, \beta_{j+1}, \dots, \beta_n)' \quad (3.23)$$

$$\sigma_j^2 = \frac{\tau^2 \sigma_a^2}{\tau^2 v_{T_j-j}^2 \delta_j + \sigma_a^2} \quad \text{with } v_{T_j-j}^2 = (1 + \Phi_1^2 + \dots + \Phi_{T_j-j}^2) \quad (3.24)$$

$$\beta_j^* = \frac{\sigma_j^2}{\sigma_a^2} [e_j(0) - \Phi_1 e_{j+1}(0) - \dots - \Phi_{T_j-j} e_{T_j}(0)]. \quad (3.25)$$

The conditional posterior distribution of α only depends on the vector δ

$$\alpha | \delta \sim \text{Beta}[\gamma_1 + m, \gamma_2 + n - i - m] \quad (3.26)$$

where $m = \sum_{t=i+1}^n \delta_t$.

This study solve problems with isolated outliers and since they come with patchy outliers it fails. Justel et al. in [49] improved the Gibbs sampling based procedure described above to identify the beginning and end patchy outliers in AR process. They proposed two Gibbs runs, the first run consists of the application of Gibbs sampling to the data in order to identify outliers. The second run will be applied to the results of the first one. It consists of two methods :

1. Adaptive method to treat identified outliers and to estimate their sizes when they are patchy.
2. Block interpolation method to reduce possible masking effects.

The specification of the location and the block size of outlier patches is done by using a window of large $2hp$ around the outlier identified by the first run. Let $c_1 \in (0, 1)$ be a critical value, typically $c_1 = 0.5$, and any observation with posterior probability greater than c_1 would be identified as outlier. We must select an other critical value $c_2 \leq c_1$ to avoid the masking effects and to identify the beginning and the end of a patchy outlier between the identified outliers. The proposed algorithm to specify outlier patches as given in [49] is as follows :

1. Choose c_1 and identify isolated outlying observations y_{t_i} using posterior probability of y_{t_i} greater than $c_1 \left(\hat{p}_{t_i}^{(s)} > c_1 \right)$. Denote the time indexes of identified outliers by $\{t_1, \dots, t_n\}$.
2. Specify h and c_2 where $c_2 \leq c_1$. For each y_{t_i} , identify a possible outlier patch $(y_{t_i-k_i}, \dots, y_{t_i+v_i})$ using the procedure mentioned above with c_2 and hp window.
3. Merge any overlapping or consecutive tentative patches.
4. If the total number of outliers is greater than $n/2$ then increase c_2 and go to step 2. If increasing c_2 cannot sufficiently reduce the total number of outliers increase h and go to step 2. If the total number of outliers is less than $\frac{n}{2}$, stop.

3.3 Level change detection

As seen in section 2, phase space reconstruction was used by [4] to detect outliers. It was also used to detect level change. In fact when a level change occurs at time t , all observations y_{t+1}, \dots, y_n lie outside the original ellipse and form a new one.

Median filtering [84] also used to detect level change. It's principle is to move a window of large h over the series and to take the median of the observations between it's limits. This method decreases the magnitude of outliers and make the series smoother and so we can detect the level change.

Gibbs Sampling was used also to detect level change in autoregressive process [55]. Let $\{y_t\}$ be a set of data points observed, say

$$y_t = \mu_t + x_t \quad (3.27)$$

where $\{x_t\}$ is an autoregressive process of order p , such that

$$\Phi(B)x_t = \Phi_0 + a_t \quad (3.28)$$

and

$$\mu_t = \mu_{t-1} + \delta_t \beta_t. \quad (3.29)$$

δ_t is level change indicator it can take values 0 or 1.

$$\begin{cases} \delta_t = 1 & \text{if the } t\text{-th observation in the beginning of a level change} \\ \delta_t = 0 & \text{otherwise.} \end{cases} \quad (3.30)$$

We can consider $\delta = (\delta_1, \dots, \delta_n)'$ as level change indicators vector.

β_t denotes the magnitude of y_t if it is a beginning of a level change and we can consider $\beta = (\beta_{i+1}, \dots, \beta_n)'$ as a level change magnitude vector.

Assume that the first level change occurs at time $t = i$ so we can say that the level change indicator vector becomes

$$\delta = (\delta_{i+1}, \dots, \delta_n)' \quad (3.31)$$

and the outlier magnitude vector becomes

$$\beta = (\beta_{i+1}, \dots, \beta_n)' \quad (3.32)$$

and the observed series becomes

$$y_t = \mu + x_t \quad t = 1, \dots, i \quad (3.33)$$

with

$$\mu = \mu_t = \mu_{t-1} = \dots \text{is a constant} \quad (3.34)$$

and

$$y_t = \mu_{t-1} + \delta_t \beta_t + x_t \quad t = i + 1, \dots, n \quad (3.35)$$

- Prior distributions

assume that

$$\Phi = (\Phi_1, \dots, \Phi_p)' \quad (3.36)$$

and

$$p(\delta_t = 1) = \alpha \quad (3.37)$$

it seems that each time point y_t has the same prior probability α of being an outlier for $t = i+1, \dots, n$.

Hence, we can say that the prior distributions are :

$$\Phi \sim N(\Phi_0, \Omega^{-1}) \quad (3.38)$$

$$\delta_t \sim \text{Bernoulli}(\alpha); \quad t = p + 1, \dots, n \quad (3.39)$$

$$\beta_t \sim N(0, \tau^2) \quad (3.40)$$

$$\alpha \sim \text{Beta}(\gamma_1, \gamma_2) \quad (3.41)$$

$$\frac{v\lambda}{\sigma_a^2} \sim \chi_v^2 \quad (3.42)$$

assume that $\Phi_0, \Omega, \lambda, v, \gamma_1, \gamma_2$ are known.

- Posterior distributions

The conditional posterior distribution of the AR parameter vector Φ as given by

$$\Phi|y, \sigma_a^2, \delta, \beta \sim N(\Phi^*, \Omega_*^{-1}) \quad (3.43)$$

$$\text{where } \Omega_* = \frac{\sum x_{t-1}x'_{t-1}}{\sigma_a^2} + \Omega; \quad \Phi^* = \left(\frac{\sum x_{t-1}x'_{t-1}}{\sigma_a^2} + \Omega \right)^{-1} \left(\frac{\sum x_{t-1}x'_{t-1}}{\sigma_a^2} \hat{\Phi} + \Omega \Phi_0 \right)$$

The conditional posterior distribution of the innovational variance σ_a^2 is

$$\sigma_a^2|y, \Phi, \delta, \beta \sim \text{inverted } -\chi^2 \quad (3.44)$$

The conditional posterior distribution of $\delta_j = (\delta_j, \delta_{j+1}, \dots, \delta_{j+k-1})$ the subvector of δ with k elements has a probability

$$p(\delta_j|y, \Phi, \sigma_a^2, \delta_{(j)}, \beta, \alpha) \propto \exp \left\{ -\frac{\sum_{t=j}^n a_t(\delta, \beta, \Phi, y)^2}{2\sigma_a^2} \right\} \alpha^l (1-\alpha)^{k-l} \quad (3.45)$$

where $\delta_{(j)}$ subvector of δ without δ_j ; $l = \sum_{t=j}^{j+k-1} \delta_t$.

$T_j = \min(n, j+i)$ and $e_t(\delta_j) = x_t - \Phi_0 - \sum_{q=1}^i \Phi_q x_{t-q}$ is the residual at time t .

The posterior distributions of the level change magnitudes β_j for $j = i+1, \dots, n$ are as follows :

- i) If $\delta_j = 0$, then there is no information about β_j and the posterior distribution will be the same that the prior one :

$$\beta_t \sim N(0, \tau^2) \quad (3.46)$$

- ii) If $\delta_j = 1$, then there is information about β_j and the conditional posterior distribution :

$$\beta_t \sim N(\beta_j^*, \sigma_j^2) \quad (3.47)$$

$$\beta_j^* = \frac{\tau^2 \sum_{t=j}^{j+i-1} c_{t-j} z_t + (n-j-i+1)c_i \bar{z}}{\sigma_a^2 + \tau^2 \sum_{t=j}^{j+i-1} c_{t-j}^2 + (n-j-i+1)c_i^2} \quad (3.48)$$

$$\bar{z} = \sum_{t=j+i}^n \frac{z_t}{n-j-i+1} \quad (3.49)$$

$$z_t = (y_t - \mu_t^*) - \sum_{p=1}^i \Phi_p (y_{t-p} - \mu_{t-p}^*) \sim N(c_{t-j} \beta_j, \sigma_a^2) \quad (3.50)$$

where

$$c_l = \begin{cases} 1 & \text{if } l = 0 \\ 1 - \sum_{j=1}^l \Phi_j & \text{for } l = 1, \dots, i \\ c_i & \text{if } l > i \end{cases} \quad (3.51)$$

The conditional posterior distribution of α only depends on the vector δ

$$\alpha|\delta \sim \text{Beta}[\gamma_1 + m, \gamma_2 + n - i - m] \quad (3.52)$$

where

$$m = \sum_{t=i+1}^n \delta_t. \quad (3.53)$$

3.4 Change point detection

[22] proposed a combination of different filters so-called Multiprocess Kamlman Filter for change-point detection. They present an algorithm which allows to estimate the probability that a changepoint has occurred, it is stable even in presence of outliers. They tried it in practice with an ECG measurements taken every 5 seconds, from a patient undergoing skin transplantation. It presents a good potentiality by detecting all changepoints and outliers very fast. So they conclude that it is adequate and reliable for on-line applications.

[21] suggested the use of two criteria which are the most important for the quality of a change-point detection algorithms such as:

1. the mean delay time τ between the changepoint appearance and it's detection.
2. the average run length which is the mean waiting time between two false alarms.

3.5 Models for on-line monitoring and forecasting medical time series

Some authors concentrated their studies on defining models for on-line monitoring time series in order to facilitate statistical analysis and forecasting variables in the course of time.

[45] proposed to use ARIMA-models. This study investigated two applications in intensive care medicine. Nineteen patients (13 males and 6 females, mean age: 52 years) were included in the first application. After two hours of liver resection surgery measurements of lactate levels and some other lab variables were started. All values were measured every 12 hours and in the second application with 25 patients (20 males and 5 females, mean age: 58 years) having severe acute respiratory distress syndrome after major gastrointestinal surgery. After the beginning of acute respiratory distress syndrome the following measurements were recorded every hour : arterial and mixed venous blood gas analysis, heart rate, arterial and pulmonary arterial pressure, central venous pressure, cardiac output, ventilatory variables.

ARIMA-models were developed after 12 hours of measurements in acute respiratory distress syndrome and after seven days of measurements in liver resection. Initial values of parameters p and q were derived from the autocorrelation function and the partial autocorrelation function. But d was found after differentiations in case of non stationarity. The final values were calculated by using Marquardt's algorithm. In general a simple model with p and $q \leq 2$ is sufficient. In case more than one model appears

satisfactory, there is some criteria such as Goodness-of-fit Criterion, Akaike Information Criterion (AIC) and Schwartz Bayesian Criterion (SBC) were computed and were useful to choose between different models.

In the application of liver resection patients, for the forecasting period of 24 hours, the values forecast by the model (including 95% confidence interval) were compared to the lab values. In the case of a good forecasting, the new values were integrated into the model and thus the period will be enlarged and the model improved progressively.

For the liver resection patients, $ARIMA(1, 0, 0)$ -model was used in 12 cases , $ARIMA(2, 0, 0)$ -model in 5 cases , $ARIMA(1, 1, 0)$ -model in one case and $ARIMA(0, 0, 0)$ -model in the last case. In all these cases lactate levels whatever be pathological or normal values did never surpass the 95% confidence interval.

For the patients suffering from acute respiratory distress syndrome, 22 cases were described by $ARIMA(1, 0, 0)$, one case required $ARIMA(2, 0, 0)$ and for the rest a linear model was used.

Later, Bauer et al. in [4] proposed a new graphical approach for pattern recognition in univariate time series from online monitoring systems in intensive care. This approach is based on phase space reconstruction. The chronological observations are combined in order to show the movement through space. The dependence structure can be clearly recognized by the elliptic form of the vector cloud. Typical disturbances of a time series like outliers, level changes and trends can be visualized by phase space reconstruction, too. This approach has been successfully applied to controlled clinical studies but not yet generalized to on-line monitoring. There are some research studies aiming to generalize this approach to multivariate time series and combine it with procedures of dimension reduction.

[36] compared several statistical methods in order to do a classification between the most important kinds of states of physiological time series. They aim at distinguishing between the occurrence of outliers, level change or trends in physiological monitoring data. They extend a case study described in [46]. On the surgical intensive care unit, on-line monitoring data was acquired from 19 critically ill patients (8 female and 11 male, mean age 65 years) with extended hemodynamic monitoring requiring pulmonary artery catheters, in one minute intervals from a standard clinical information system. These data was stored into SQL database and treated via a standard statistical software.

From a total of 550.000 single observations of 7 variables, segments of 150 to 500 observations for each variable were visually classified into 5 patterns :

1. presence of outlier,
2. temporary level change,
3. permanent level change,
4. no change,
5. trend.

AR(2) model	Phase space model	Dinamic linear model
All series with outliers are identified,	All series with outliers are identified,	All series with outliers are identified with estimation interval of 60 minuts,
All series with level changes are identified,	Identification of level changes failed,	Level changes were detected best by moving an estimation interval of 30 minuts through the series.
All series without change are identified,	All series without change are identified,	All series without change are identified with estimation interval of 60 minuts,
Trend detection failed.	Trend detection failed.	

Table 3.1: Results of segments analysing

For a total of 134 time series : 23 were classified without change, 35 as containing change, 24 as containing temporary level change, 42 as containing permanent level change and 10 as showing a trend pattern. The segments were analyzed with $AR(2)$ -models phase space models and dynamic linear models and they obtained the following results in table 3.1.

3.6 Multivariate time series and dimension reduction

Most of these studies solve the smoothing problems but fail when we introduce high dimensional real patient data (which are present as multivariate time series [4]). Some proposals and much progresses have been made to convert these models and techniques to deal with this kind of series. But the multivariate time series structure, which presents a lot of directions for deviations from the steady state system and a multitude of relations between the variables [39] cause their failure. Recently several authors thought that it may be more efficient if they try other techniques for multivariate time series. Graphical interaction models have been developed for analyzing these relations mentioned above.

In fact, we can represent multivariate time series by a graph which contains vertices symbolizing the random variables and a multitude of edges between them symbolizing the relations that exist between the random variables. Let a graph $G = (V, E)$ consists of a finite set of vertices V and a set of ordered pairs of vertices so-called edges set $E \subset V \times V$. If a and b a pair of connected vertices were visualized by a circle for each vertex and an undirected edge connecting them (see fig.3.2).

this means that $(a, b) \in E$ and $(b, a) \in E$. But if only $(a, b) \in E$ then a directed edge is drawn from a to b (see fig.3.3).

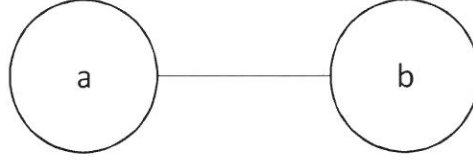


Figure 3.2: Undirected edge

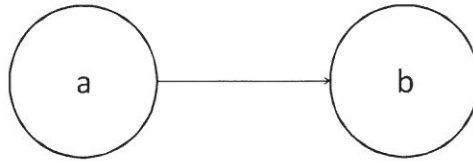


Figure 3.3: Directed edge

In reality further types of associations can be present in a graph. There are direct and induced associations, strength and weak associations etc. In statistical meaning, these associations represent the conditional dependence of the corresponding variables and a missing one represents the conditional independence. Following the Pairwise Markov property we can say that two vertices a and b may be indirectly connected or connected by a path if there exist a_1, \dots, a_k $k \geq 0$ such that all $(a, a_1), (a_1, a_2), \dots, (a_{k-1}, a_k), (a_k, b) \in E$ and $(a, b) \notin E$ and following Global Markov property having $A \subset V, B \subset V$ and $C \subset V$. we say that C separates A and B which are conditionally independent if any path between two variables $a \in A$ and $b \in B$ passes necessarily by $c \in C$. This property means that if we control the effects of the separating set C then A and B will be not associated hence we can control better the system.

These models are developed for analyzing the relations between the components of multivariate time series. For describing the interactions between the components of a multivariate time series they used a partial spectral coherence as [20], which consists to measure the linear dependence between two components of a multivariate time series after removing all the effects of the other variables [14]. [20] proposed a method by combining this tool with the concept of graphical models in order to detect the relations between the variables of a multivariate time series. Gather and Imhoff in [39] adopted this technique to hemodynamic system of critically ill patients monitored in intensive care.

The biggest problem of multivariate time series techniques is the high-dimensionality. For this purpose there were trials dealing with the so-called "*dimension reduction*" which consists of reducing the number of variables without loss of information. Li in [53] proposed the following dimension reduction model :

$$y = g(\beta'x, \varepsilon) = g(\beta'_1x, \dots, \beta'_kx, \varepsilon) \quad (3.54)$$

where :

y is a real-valued response variable,

x is the p -dimensional random vector of explanatory variables,

and ε is a random error.

g unknown link function. $g : \mathfrak{R}^{k+1} \mapsto \mathfrak{R}$,

β is a p by k matrix with columns β_i ($i = 1, \dots, k$), $k < p$ called effective dimension reduction space,

β_1, \dots, β_k p -dimensional vectors called effective dimension reduction (edr) directions have to be estimated from the data.

To forecast y we need to know $\beta'x$. Denote $S(\beta)$ the space spanned by the columns of the matrix β so-called dimension reduction sub-space. The idea is to reduce the dimension of the forecasters from p to k if we know β and find the smallest dimension reduction sub-space. But it is not always unique, that's why we must find the central dimension reduction sub-space which is unique when it exists. Following [87, definition 1] :

A sub-space S is a central dimension reduction sub-space for the regression of y on x if:

- (a) S is a dimension reduction subspace and
- (b) $S \subset S_{dr\ s}$ for all dimension reduction sub-spaces $S_{dr\ s}$, i.e. $S = \cap S_{dr\ s}$.

It will be denoted by $S_{y|x}$.

The prominent technique for the dimension reduction is the Sliced Inverse Regression [38]. This technique proposes the following algorithm to estimate edr-directions [38, p2]:

1. Standardization:

$$z_i = \hat{\Sigma}^{-\frac{1}{2}} (x_i - \bar{x}), \quad i = 1, \dots, n \quad (3.55)$$

where $\hat{\Sigma}$ and \bar{x} denote the sample covariance matrix and the sample mean of x_i , $i = 1, \dots, n$, respectively.

2. Slicing :

Divide the range of y into H slices I_1, \dots, I_H . Let n_h be the number of observations of y which fall into I_h , $h = 1, \dots, H$, and let $p_h = \frac{n_h}{n}$.

3. Sample mean-vector within each slice :

$$\hat{m}_h = \sum_{y_i \in I_h} \frac{z_i}{n_h}. \quad (3.56)$$

4. Principle Component Analysis (PCA) :

Let

$$\hat{V} = \sum_{h=1}^H p_h \hat{m}_h \hat{m}_h' \quad (3.57)$$

and compute the eigenvalues $\hat{\lambda}_1 \succeq \dots \succeq \hat{\lambda}_p$ and corresponding eigenvectors η_1, \dots, η_p of \hat{V} with

$$\hat{\eta}_i' \hat{\eta}_j = \delta_{ij} \quad (3.58)$$

where δ_{ij} denotes the Kronecker symbol.

5. Retransformation :

The edr directions are estimated by

$$\hat{\beta}_k = \hat{\Sigma}^{-\frac{1}{2}} \hat{\eta}_k, k = 1, \dots, K. \quad (3.59)$$

3.7 Conclusion

In this chapter, and after a large literature review we can see that :

- The individual model identification and statistical evaluation of a single patient is used and seems to constitute a successful methodology in intensive care monitoring.
- The use of ARIMA models for the analysis of univariate intensive care time series appears to be very promising. Those models are used both for outlier detection and level change detection and seem to have satisfactory results [45], [32].
- The pattern detection in multivariate time series is more difficult than in univariate series since there are a big number of directions for deviations from the steady state [37].

Chapter 4

ONLINE MULTIAGENT MONITORING SYSTEM FOR INTENSIVE CARE UNITS

Principal results of this chapter are published in the 7^{ème} Journées Scientifiques des Jeunes Chercheurs en Génie Electrique et Informatique (GEI'2007) [63], and the proceedings of the 7th Tunisia-Japan Symposium on Society, Science & Technology (TJSSST 2006) [62].

4.1 Introduction

Physiological data altogether form a complex system. Therefore monitoring such variables inherits all the necessary characteristics to be modeled with the Multi-Agent System (MAS) approach. These characteristics are modularity, interconnections and parallel processing [28]. Multi-agent approach seems to be the only technique which can help to reduce the complexity of the system by creating modular components, which solve private subtasks that constitute together the hole goal [77]. Every agent utilizes its own technique for solving the subtask. In case the subtasks are interconnected, agents have to communicate to each other. The system can be represented as a community of intelligent entities, solving the private tasks and constituting a whole complex system.

On the other hand, monitoring and quick anomaly detection are essential to ICU staff. Quick detection refers to the speed of data that move rapidly through the ICU information system, from patient's connected sensors to anomaly detection component, to alarm generation and finally to medical staff. Considering a patient critical care requiring quick response, the monitoring system must quickly detect anomalies to allow medical staff to make the appropriate decision concerning treatment. The aim is to minimize health cost, to enhance care quality, to lower patient hospitalization time and to improve medical staff work quality. In this context, a real-time method must be proposed to improve the quick response capability.

In this chapter, we present a hybrid multi-agent system to solve the patient's critical state monitoring problems in real time. The approach allows the integration of intelligent entities with time series technology. This system will be called Online Multi-Agent Monitoring System (MAMS).

4.2 System description

Our system is created by a three layered MAS (see Fig.4.1), which is aimed to provide physiological data, to analyze them and to generate alarm signals.

Following the architecture pattern presented in chapter 1, the proposed system is composed of :

- an acquisition System having a sensor role,
- a monitoring System having a transformer role,
- and an alarm System having a deliverer role.

The architecture of each component are described as follows (Fig.4.1):

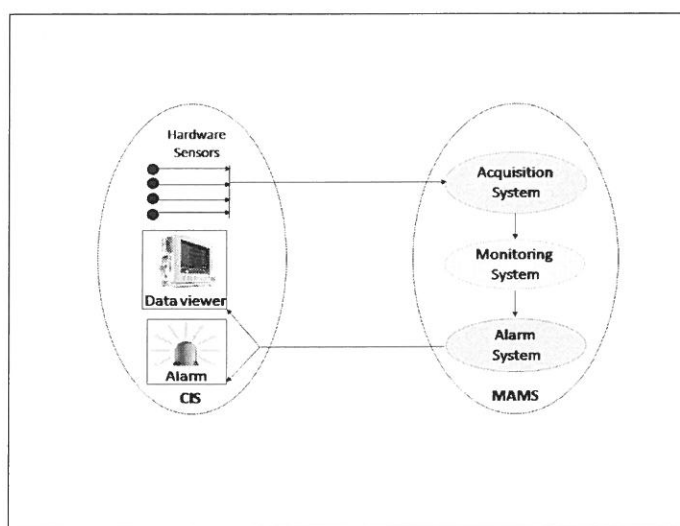


Figure 4.1: Global MAMS Architecture

4.2.1 The Acquisition System Architecture

The Acquisition System is an agent-based system (see Fig. 4.2), it integrates static and reactive agents working in parallel. Each agent, called Acquisition Agent, collects data from the specific device.

The Acquisition Agent is composed of (see Fig. 4.3):

- a sensor called Digitizer that perceives data coming from the patient,
- a transformer called Time Series Converter that converts data into regularly spaced time series,

- a deliver called Time Series Dispatcher that dispatches deduced time series to the Monitoring System.

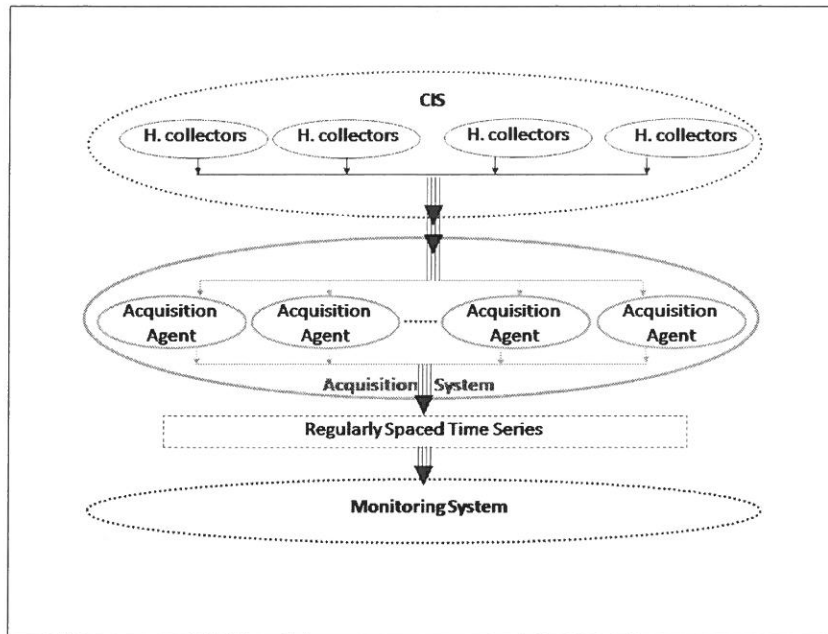


Figure 4.2: The Acquisition System Architecture (a)

4.2.2 The Monitoring System Architecture

The Monitoring System is composed by a multitude of dynamic subsystems called Time Series Monitoring Systems (TSMS) (see Fig. 4.4). Each TSMS has a dynamic and cognitive agent called Control Agent and a static and reactive agent called Graphical Agent (see Fig. 4.5).

The Control Agent Architecture

The Control Agent is composed of:

- a sensor called Time Series Receiver that receives time series from the acquisition agent,
- a transformer called Pattern Detectors that detects series patterns as outliers, trends, level changes...
- a deliverer called Anomalies Deliverer that sends alarm requests to the alarm system.

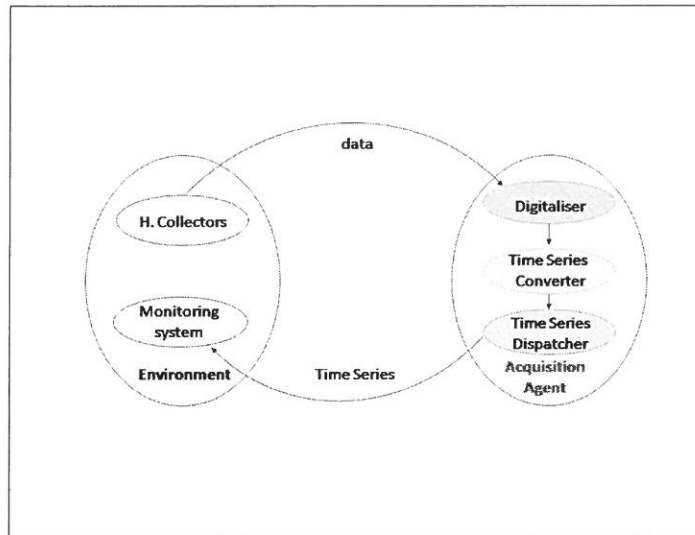


Figure 4.3: The Acquisition System Architecture (b)

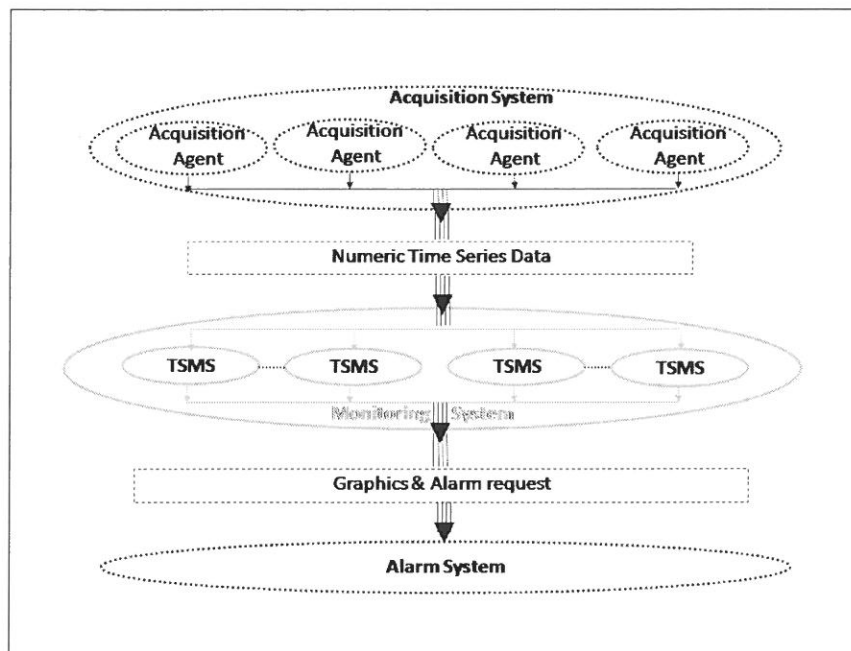


Figure 4.4: The Monitoring System Architecture (a)

The Graphical Agent Architecture

The Graphical Agent is composed of:

- a sensor called the Time Series Receiver,
- a transformer called the graphical transformer that converts time series into graphical models,
- a deliverer is the graphics displayer that displays graphics on the bedside data viewer.

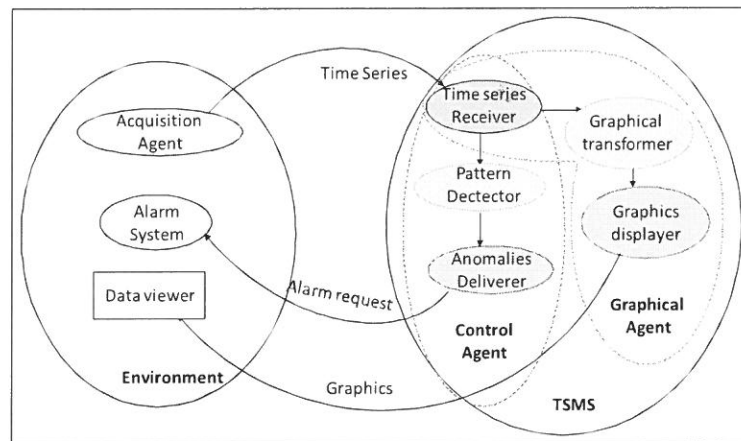


Figure 4.5: The Monitoring System Architecture (b)

4.2.3 The Alarm System Architecture

The Alarm System adopts the same pattern for its architecture as illustrated by Fig. 4.6 and Fig. 4.7.

The Alarm System receives data from the Monitoring System and exactly from Control Agents. The Alarm System embodies one static and reactive agent that works autonomously: the Alarm Agent.

The Alarm Agent collects alarm requests received through the Alarm receiver, decides about these requests and activates alarm through the Loudspeaker.

4.3 The System's working principles

As seen above, MAMS is a three layered MAS. Each layer is in turn an MAS which communicates with other layers by Agent Communication Language (ACL) messages (see Appendix A). In the following, we present the algorithm of each agent in the different system's layers.

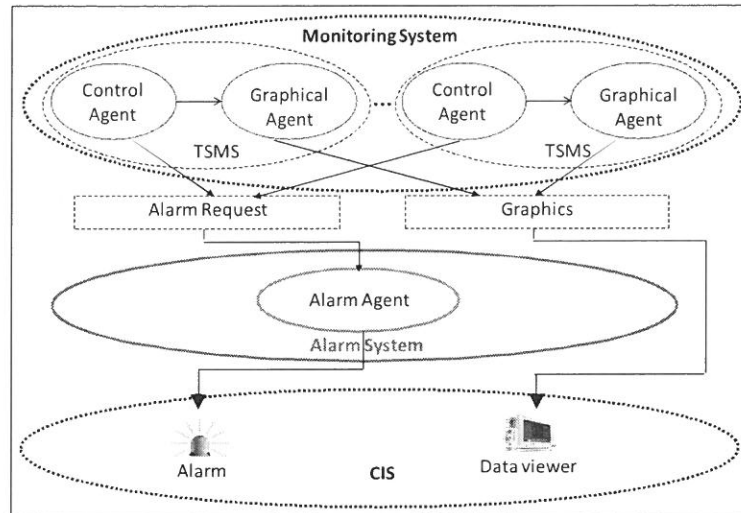


Figure 4.6: The Alarm System Architecture (a)

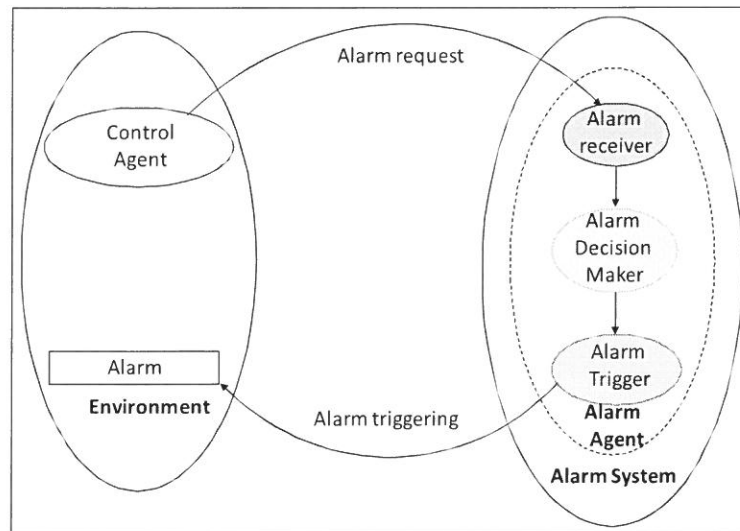


Figure 4.7: The Alarm System Architecture (b)

StartingAgent Algorithm

A starting Agent collects physiological variables names as arguments and puts them in *argv* table. It skims through the table and creates an Acquisition Agent for each variable. After that, it creates the Alarm Agent and finally it destroys itself (see Algorithm 1).

Algorithm 1 Starting Agents Algorithm

1. Algorithm StartingAgent(argv [])
 2. Var *i* : integer
 3. **Begin**
 4. argumentsRecuperation () ;
 5. **for** *i* = 1 to length(argv) **do**
 6. Create and start up an agent instance of AcquisitionAgent class;
 7. **end for**
 8. Create and start up agent instance of AlarmAgent class;
 9. doDelete () ; // *StartingAgent will destroy itself after achievement of his tasks.*
 10. **End**
-

AcquisitionAgent Algorithm

The AcquisitionAgent is a reactive agent and an instance of AcquisitionAgent class. It has the following algorithm (see Algorithm 2):

Algorithm 2 Acquisition Agents Algorithm

1. Algorithm AcquisitionAgent(argv [])
 2. **Begin**
 3. argumentsRecuperation () ;
 4. Create and start up agent instance of ControlAgent class;
 5. Create and start up agent instance of GraphicalAgent class;
 6. AddBehaviour(AcquisitionAgentBehaviour);
 7. **End**
-

The Acquisition Agent whose name is received as argument is started up. It creates its corresponding Control Agent and Graphical Agent and acquires its own behavior as presented in Algorithm 3:

Algorithm 3 Acquisition Agents Behavior Algorithm

```

1. Algorithm AcquisitionAgentBehavior ()
2.   Var
3.     s: sensor
4.     val : integer
5.     rcvCtrl,rcvAff : String
6. Begin
7.   rcvCtrl=constructControlAgentName() ;
8.   rcvAff= constructGraphicalAgentName() ;
9.   while (not endOfRead(s)) do
10.    val=read(s) ;
11.    sendMessage(val,rcvCtrl) ; // Sending val by message to the rcvCtrl agent.
12.    sendMessage(val,rcvAff) ; // Sending val by message to the rcvAff agent.
13.    sleep(n) ; // pause of n milliseconds.
14.  end while
15.  doDelete();
16. End

```

The Acquisition Agent reads data from the sensor each n milliseconds (n is a parameter introduced by the staff) and sends them to its corresponding Control Agent and Graphical Agent. When the sensor is disconnected and no data is read the Acquisition Agent will destroys itself.

ControlAgent Algorithm

It's a cognitive agent which instantiates the ControlAgent class and is a part of Monitoring System. It has the following algorithm (Algorithm 4):

Each n milliseconds, the Control Agent receives a message having data from the Acquisition Agent. It tests the presence of abnormal or critical state using a monitoring technique. In the presence of 5 consecutive critical data, the Control Agent sends an Alarm request to the Alarm Agent. Otherwise, it sends a Cancel request. When the Acquisition Agent is destroyed and stops sending messages, the Control Agent will destroy itself.

GraphicalAgent Algorithm

The Graphical Agent is a reactive agent which instantiates the GraphicalAgent class and is a part of the Monitoring System. It has the following algorithm (Algorithm 5):

Each n milliseconds, the Graphical Agent receives a message from Acquisition Agent having the

Algorithm 4 Control Agents Algorithm

```
1. Algorithm ControlAgentBehavior ()
2.   Var
3.     msg: message
4.     content, nbrErr : Integer
5.     Cancel: boolean = false
6. Begin
7.   msg=recieveMessage() ;
8.   while (msg/ = Null) do
9.     content=contentRecuperation(msg) ;
10.    if (abnormal(content)) then
11.      nbrErr = nbrErr + 1 ;
12.    else
13.      nbrErr = 0 ;
14.      if (Cancel) then
15.        sendMessage ("Cancel ", alamAgent) ;
16.        RAZ = faulse ;
17.      end if
18.    end if
19.    if (nbrErr = 5) then
20.      Cancel = true ;
21.      sendMessage (" Alarm ", Agent_alarme) ;
22.    end if
23.    sleep (n) ; // pause of n milliseconds.
24.    msg=recieveMessage() ;
25.  end while
26.  doDelete();
27. End
```

Algorithm 5 Graphical Agents Algorithm

```

1. Algorithm GraphicalAgentBehavior ()
2.  Var
3.  msg: message
4.  content : integer
5. Begin
6.   msg=recieveMessage () ;
7.   while(msg/ = Null) do
8.     content=contentRecuperation(msg) ;
9.     drawGraphic(content) ;
10.    sleep (n) ; // pause of n milliseconds.
11.    msg=recieveMessage () ;
12.   end while
13.   doDelete();
14. End

```

same content as the Control Agent. The graphical Agent ads this content to its corresponding graph. When the Acquisition Agent is destroyed and stops sending messages, the Graphical Agent will destroy itself.

AlarmAgent Algorithm

The Alarm Agent which is an instance of AlarmAgent class, is a cognitive agent and has the following algorithm (Algorithm 6) :

Each *n* milliseconds, the Alarm Agent receives messages from all the Control Agents. Each Alarm request causes the addition of the Control Agent Name in the Alarm list and each Cancel request causes the removal of the Control Agent Name from the Alarm list. After that, and using its rules base, the alarm Agent decides about triggering alarm or not.

4.4 Conclusion

In this chapter an approach to develop an agent-based monitoring system has been introduced. The software agents use time series technology for anomaly discovery, which will be used as a foundation for alarm generation.

Based on the observation that UML doesn't provide currently sufficient means for the design of MAS, we have used AUML formalism. In the specification phase we identified agents and their relationships in the actor's model, goals and roles of each agent in the agent goal diagrams and the agent interaction with its environment in the use case and activity diagrams. In the design phase we used class

Algorithm 6 Graphical Agents Algorithm

```
1. Algorithm AlarmAgentBehavior()
2.   Var
3.     msg : message
4.     content : integer
5.     delay : integer
6.     nbCtrl : integer
7. Begin
8.   nbCtrl = ControlAgentNumber;
9.   while (true) do
10.    for each ControlAgent do
11.      msg=recieveMessage() ;
12.      if (msg = Null) then
13.        nbCtrl - -;
14.      end if
15.      content=contentRecuperation(msg) ;
16.      if (content = "Alarm") then
17.        addMessageSourceToAlarmSourceList() ;
18.      else
19.        if(content= "Cancel") then
20.          deleteMessageSourceFromAlarmSourceList() ;
21.        end if
22.      end if
23.    end for
24.    if (nbCtrl = 0) then
25.      doDelete();
26.    end if
27.    rulesBaseConsultation();
28.    sleep(n);
29.  end while
30. End
```

diagram. For modeling dynamic behavior, a sequence diagram is used. Finally we presented details on algorithms.

Chapter 5

EXPERIMENTAL STUDY

Principal results of this chapter are published in the proceedings of the 4th International Multi-Conference on Systems, Signals & Devices (SSD-07) [65], the proceedings of the 24th Annual Conference and Exhibition HEALTHCARE COMPUTING (HC2007) [64], as poster in the International Symposium on Computational Biology & Bioinformatics (ISBB06) [60] and will appear in proceedings of International Colloquium on Computing, Communication, Control, and Management (CCCM 2010) [66].

5.1 Introduction

This chapter outlines the range of experiments for the proposed monitoring system, and details the experimental cycles. It covers the nature of our experiments, and describes the aims of these. The main objectives of our experiments are the evaluation of the following.

1. **The Control Agent performance** : the Control Agent performance is measured by monitoring one variable at a time. In order to assess the performance, sensitivity and specificity must be measured. The sensitivity could be measured by comparing the number of detected anomalies versus the number of missed ones. For this purpose we define the ratio *CtrlAgtSensitivity* as:

$$\textit{CtrlAgtSensitivity} = \frac{\textit{AnomaliesDetected}}{\textit{Anomalies}} \quad (5.1)$$

The specificity is measured by the number of significant anomalies detected (TPAn) versus irrelevant anomalies detected (FPAn). It's the ratio of significant anomalies detected to all significant ones.

$$\textit{CtrlAgtSpecificity} = \frac{\textit{TPAn}}{\textit{TPAn} + \textit{FPAn}} \quad (5.2)$$

- 2 **The Alarm Agent performance** : the Alarm Agent performance is measured by monitoring several variables at a time. In this case, the sensibility is measured by the number of significant generated alarms (TP) versus the number of not generated ones. It's the ratio of significant

generated alarms to all significant ones.

$$AlAgtSensitivity = \frac{TP}{Significant\ Alarms} \quad (5.3)$$

The specificity is measured by the number of significant generated alarms (TPA) versus irrelevant generated alarms (FP). It is defined as the ratio of true positive alarms to all generated ones.

$$AlAgtSpecificity = \frac{TP}{TP + FP} \quad (5.4)$$

3. **The global system performance:** the global system performance is inferred from the Alarm Agent performance (i.e. from its sensitivity and specificity). In addition to assess the capability of the global system, the time elapsed, between the anomaly detection and the alarm generation, must be calculated.

5.2 Control Agent functionality

The main objective of Control Agent is to detect anomalies. Anomaly detection assumes that the critical state will always reflect some deviations from normal patterns [18]. Hence, the Control Agent will be in charge of fitting in the data and detecting the deviations from the normal pattern.

To evaluate the Control Agent functionality, we need to integrate one of the monitoring techniques in the core of the agent to replace the behavior rules. In fact, the agent functioning is described by a set of behavior rules which are similar to production rules. The main disadvantage of the rules based detection systems is that they only detect the anomalies for which they were trained. Novel anomalies, unknown anomalies or even variants of common anomalies often go undetected [18].

There are several monitoring techniques that could be used in our work:

1. **Fixed threshold based technique :** each agent uses fixed thresholds defined previously by the physician. The contribution of fixed threshold based system in comparison with the usually used system is the presence of agents which solve the interdependence problem.
2. **Traditional and modern statistical machine learning techniques:** unlike rule based techniques, some machine learning based systems can detect novel, unknown and variants of known anomalies. Here are some examples of time series and machine learning techniques.

(a) *Time Series technology.* As seen in chapter 3, time series technology is widely used in medical data monitoring considering the serial correlation between the physiological observations [32]. Especially Autoregressive (AR) models [36], [4] and Autoregressive Integrated Moving Average (ARIMA) models which are the most adequate to medical data as [45] and the most often used time series models as [32].

(b) *Artificial Neural Networks*. Neural networks have the advantage of easier representation of non-linear relationships between input and output but the defects are that their computational load is very heavy [18].

(c) *Support Vector Machines*. Support Vector Machines are learning based algorithms designed for binary classification and regression [50]. Support Vector Machines have proven to be a good candidate for anomaly detection because of their speed and scalability [18].

Both Artificial Neural Networks and Support Vector Machines have the disadvantage of being designed for a supervised learning. This means that the classification is based on labeled training input vectors. But in ICUs, data are not labeled. Hence, those techniques can't be used in intensive care monitoring.

In this chapter, only Fixed threshold and time series based techniques are retained.

Relating to time series techniques, we rely on results and conclusions derived in [45] and [32]. The finding of these studies show that ARIMA models appear to be very promising for intensive care time series analysis. We follow the same approach.

The primary goal of our system is to be automatic. Hence, both modeling steps (identification and estimation) and anomalies detection have been be totally automated.

5.3 The ARIMA model

ARIMA-models have proven themselves to be relatively robust and very promising for medical time series analysis. Imhoff et al. in [45] present some reasons for that.

ARIMA-models are flexible and applied to a wide spectrum of time series analysis. The abbreviation ARIMA stands for Auto Regressive Integrated Moving Average. The general $ARIMA(p, d, q)$ -model is a combination of three processes: Autoregression (AR), differentiation in order to eliminate integration (trend) of a time series (I), and moving average (MA).

5.3.1 Autoregressive (AR) process

A stochastic process (x_t) is an autoregressive process of the order p , indicated by the notation $AR(p)$ or $ARIMA(p, 0, 0)$,

where

$$x_t = \theta_0 + \phi_1 x_{t-1} + \dots + \phi_p x_{t-p} + \varepsilon_t, \quad t \in \mathbb{N} \quad (5.5)$$

or

$$(1 - \phi_1 B - \dots - \phi_p B^p)x_t = \theta_0 + \varepsilon_t \quad (5.6)$$

or

$$\Phi(B)x_t = \theta_0 + \varepsilon_t, \quad (5.7)$$

where

$$\Phi(B) = (1 - \phi_1 B - \dots - \phi_p B^p), \quad (5.8)$$

B is the backshift or lag operator such that

$$Bx_t = x_{t-1} \quad (5.9)$$

and

$$B^j x_t = x_{t-j}. \quad (5.10)$$

$$\{\varepsilon_t\} \text{ a white noise process } \sim i.i.d. \mathcal{N}(0, \sigma_\varepsilon^2), \quad (5.11)$$

$$\theta_0 = (1 - \phi_1 B - \dots - \phi_p B^p) \mu_t. \quad (5.12)$$

$$\mu_t = E(x_t) \quad (5.13)$$

μ_t is the mean function of x_t .

In the case of stationary process, since the distribution function is the same for all t , the mean function

$$\mu_t = \mu \text{ is a constant.} \quad (5.14)$$

and ϕ_1, \dots, ϕ_p are unknown weights measuring the influence of the past observations on the current observation.

In fact, each value in an $AR(p)$ -process is determined by p preceding values plus a random disturbance. In an $AR(1)$, for example, the value at time t is a function of the preceding value at time $t - 1$, and the value at time $t - 1$ is a function of the preceding value at time $t - 2$, and so on.

$\Phi(B) = (1 - \phi_1 B - \dots - \phi_p B^p)$ is a polynomial in B of degree p . in AR model it is assumed that the polynomial $\Phi(B) = 0$ have all its roots outside the unit circle. In terms of $AR(1)$ -process it implies that

$$|\phi_1| < 1. \quad (5.15)$$

For an $AR(2)$ -process

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 = 0 \quad (5.16)$$

must lie the unit circle, which implies that the parameters ϕ_1 and ϕ_2 must lie in the triangular region

$$\phi_1 + \phi_2 < 1, \quad (5.17)$$

$$\phi_2 - \phi_1 < 1 \quad (5.18)$$

and

$$-1 < \phi_2 < 1. \quad (5.19)$$

5.3.2 Moving Average (MA) process

A stochastic process (x_t) is a moving average process of the order q , indicated by the notation $MA(q)$ or $ARIMA(0, 0, q)$,

where

$$x_t = \theta_0 + (1 + \theta_1 B + \dots + \theta_q B^q) \varepsilon_t \quad (5.20)$$

or

$$x_t = \theta_0 + \Theta(B) \varepsilon_t, \quad (5.21)$$

where

$$\Theta(B) = (1 - \theta_1 B - \dots - \theta_q B^q), \quad (5.22)$$

$$\theta_0 = \mu \quad (5.23)$$

and

$$\theta_1, \dots, \theta_q \quad (5.24)$$

are unknown weights measuring the influence of the past disturbances ε on the current observation.

Because

$$1 - \theta_1^2 - \dots - \theta_q^2 < \infty, \quad (5.25)$$

a finite moving average process is always stationary.

The difference between an autoregressive process and a moving average process is important. Each part in an autoregressive model relates the actual value to past values in a linear fashion. The moving-average model relates the actual value to the errors or disturbances of previous ones. Since these values in turn are weighted averages of the previous ones, the effect of a given disturbance in an autoregressive process dwindles as time passes. In a moving average process, a disturbance affects the system for a finite number of periods after that it ceases to affect it.

5.3.3 Integration

The stationarity of a series is necessary for the estimation of AR and MA processes. Therefore, time series that show a trend, due to a reflect of a cumulative effect of some processes, should be differenced, until stationarity is accomplished.

Differencing means to calculate the increments $(x_t - x_{t-1})$ between subsequent values as much as possible until the ARIMA model follows a stationary ARMA one. d represents the number of times we need to differentiate the time series before being stationary. So if $d = 1$, it means that we differentiated only one time and $y_t = x_t - x_{t-1}$ or $y_t = (1 - B)x_t$ is an $ARIMA(p, 1, q)$ -process and if $d = 2$, it means

that we differentiated twice, the first one, $y_t = x_t - x_{t-1}$ and the second one $z_t = y_t - y_{t-1}$. $z_t = (1 - B)^2 x_t$ is an $ARIMA(p, 2, q)$ process.

An $ARMA(p, q)$ model assumes that x_t can be presented as

$$(1 - \phi_1 B - \dots - \phi_p B^p)x_t = \theta_0 + (1 + \theta_1 B + \dots + \theta_q B^q)\varepsilon_t \quad (5.26)$$

or

$$\Phi(B)x_t = \theta_0 + \Theta(B)\varepsilon_t \quad (5.27)$$

and an $ARIMA(p, d, q)$ model assumes that x_t can be presented as

$$\Phi(B)(1 - B)^d x_t = \theta_0 + \Theta(B)\varepsilon_t. \quad (5.28)$$

5.3.4 Seasonality

The seasonality is a pattern in a series that repeats regularly. The seasonal part of an ARIMA model has the same structure as the non-seasonal part. A seasonal ARIMA model is classified as an $ARIMA(p, d, q)x(P, D, Q)_s$ model.

5.3.5 Building ARIMA models

In this section we will outline the practical steps which need to be undertaken to use ARIMA models. We will consider Box and Jenkins [13] approach which is embodied in the model-building procedure described in figure 5.1. The initial values for p, d, q (and probably P, D and Q when there is presence of seasonality) were derived from the Sample Autocorrelation Function (SACF)

$$\hat{\rho}_h = \frac{\sum_{t=1}^{N-h} (x_t - \bar{x})(x_{t+h} - \bar{x})}{\sum_{t=1}^N (x_t - \bar{x})(x_t - \bar{x})}, \quad (5.29)$$

$$\bar{x} = \frac{1}{N} \sum_{t=1}^N x_t, \quad (5.30)$$

and the Sample Partial Autocorrelation Function (SPACF) of the original series, and after some order differentiations if necessary. The SACF estimates the unknown autocorrelation function of the process. While the SPACF estimates the autocorrelation between x_t and x_{t+h} , for an $AR(p)$ process the SACF should be exponentially declining and SPACF should be about zero all the time lags $h > p$. Conversely, for an $MA(q)$ process the SACF should be about zero from time lag $q + 1$, while the SPACF should be exponentially declining. For mixed $ARMA(p, q)$ process the SACF and SPACF show more complex patterns. Their identification often takes several cycles of the procedure.

In the estimation phase, the unknown parameters $\phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$ are estimated from the data. If more than one model appeared suitable for the series, all models were estimated. Commonly we estimate SACF and SPACF of the residual series. Models were accepted only when the residual SACF and SPACF values do not exceed the 95% confidence limits. This indicates the absence of correlations between the residuals and therefore they present a white noise process.

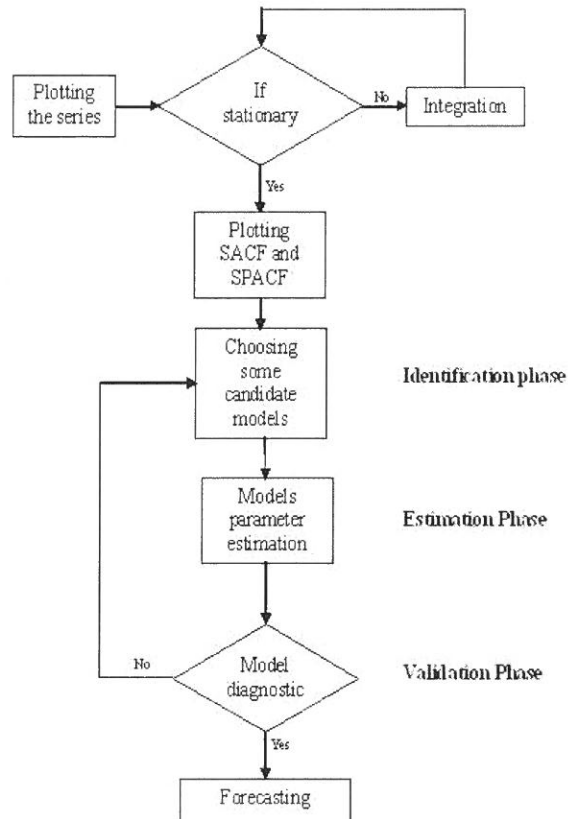


Figure 5.1: ARIMA-model building procedure

In the validation phase, for each model coefficient the t-statistics were determined to test their significance. Only models in which the coefficients were significant, were selected.

Finally, if many models are selected the Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC) will also be computed to select the more reliable one. If the model turns out to be satisfactory we can use it for further analysis such detecting anomalies in the series or forecasting subsequent values. Otherwise, we should modify it according to the impressions gained in the diagnosis stage and then iterate until a satisfactory model is derived.

A different approach is the application of several number of models where the orders vary over a wide range. The models are then compared by a calculated goodness of fit criterion and so the best model will be selected [45] .

5.4 Automation of the modeling steps

To automate the modeling steps detection we developed an identification and estimation procedures, in R language. It consists of the generation of different values of $ARIMA(p, d, q)x(P, D, Q)_s$

orders. The ARIMA model is expressed as :

$$(1 - B)^d \phi_p(B)(1 - B^s)^D \Phi_P(B^s)Z_t = \theta_q(B)\Theta_Q(B^s)a_t, \quad (5.31)$$

with

$$\begin{cases} \phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \\ \Phi_P(B) = 1 - \phi_{s1} B^s - \phi_{s2} B^{2s} - \dots - \phi_{sP} B^{Ps} \\ \theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \\ \Theta_Q(B) = 1 - \theta_{s1} B^s - \theta_{s2} B^{2s} - \dots - \theta_{sQ} B^{Qs} \end{cases} \quad (5.32)$$

The optimal order selection is based on usual goodness of fit criteria. In order to save computation time, as suggested by [36] and [45] who have pointed out that physiological time series processes are characterized by short memory, values between 0 and 3 are assigned to ARIMA parameter orders $(p, d, q)(P, D, Q)$. Akaike Information Criterion (AIC) and mean squared error Criteria are used for optimal selection of those parameters.

The selection procedure is based on the first encountered stable part of the series consisting of 60 observations.

5.5 Anomaly detection

The first 60 observations are used to identify the orders and to estimate the parameters of the ARIMA model (ϕ_s and θ_s). The selected model is then used to generate one step ahead point forecast and 95% interval forecast for period $T + 1$. This one step forecast value is compared to the observed value for the same period $T + 1$. If the newly recorded value for $T + 1$ lies within the interval forecast obtained earlier, the new value is then integrated into the updating of model parameters. Therefore, continuous updating of the sample size on which are based the model building and forecasting is performed. However, a very large estimation period can increase the computation time. Hence, a moving window of 600 observations was used.

Usually, changes of variables over time (such as level changes, trends and variance changes) are more important than a single pathological value at the time of observation (outlier). Hence, the online detection of qualitative patterns such as outliers, level changes, trends and variance changes is important for assessing the patient's state. So, only level changes, trends and variance changes need medical intervention. Outliers generate insignificant alarms.

5.5.1 Outliers' detection rule

An outlier means only one value deviates from the forecast model (outside the 95% confidence interval); in this case the forecast value will be integrated into the model instead of the observed one.

5.5.2 Level change, trends and variance change detection rules

Level change, trend and variance change mean persistence of deviations from the forecast value. In order to not consider patchy outliers as anomalies, we set the system to declare anomaly after 5 consecutive values deviating from the forecast model as in [36]. This procedure was integrated in the Control Agent behavior using Java to R Interface (JRI).

5.6 Alarm Agent functionality

To evaluate the Alarm Agent functionality, use of several patient's variables has been done. Each variable will be affected to a Control Agent. The Control Agents use the same monitoring technique.

5.7 Data description

It's essential to test the system with real data. But, in Tunisia, data is not available. This is due to the fact that in ICUs most of the time monitors don't have a recording unit. In addition there is a lack of sufficient number of monitors in ICUs and some times only one functional monitor is used for monitoring the state of many patients. Therefore, the staff shares it among all the patients and so, there is no continuous data-recording data for patients. Hence, resorting to Physiobank [40] was the obvious solution. In this site, MIMIC (Multiparameter Intelligent Monitoring for Intensive Care) database was used. This database contains data from hemodynamically unstable patients hospitalized in 1996 in ICU of the cardiology division of the Teaching Hospital of Harvard Medical School. It includes 100 patient records, each typically contains 11 hours 43 minutes and 8 seconds of continuous data recorded each second (42188 records for each physiological variable). Each recording is accompanied by detailed annotations or labels that point to specific locations and describe events at those locations.

From this base, several patient records are used to test performance of our system. Two of these patient records are presented as examples in this chapter. The peculiarity of these data is the presence of the different qualitative patterns such as outliers, level changes gradual and abrupt trends.

5.8 Experimental topology

In the first set of experiments, only Control Agents are tested. We used two kinds of monitoring techniques, the fixed thresholds technique, and the time series techniques. They will be compared to the usually used system in the ICUs. Each patient record of the MIMIC database has informations about the used thresholds. We notice, that the staff widen the tolerance interval during the monitoring period when the fixed threshold causes a lot of generated alarms. For example, in some patient records, the threshold of the Arterial Pressure which in reality must not exceed 140 *mmHg*, the staff changes it to 220 *mmHg* when it persists high without caring about what happens to the patient.

In the second set of experiments, we used the same systems, but this time, Alarm Agent will be tested. In each time, we will compare the generated alarms, with respect to their significance, their type and their detection (number of significant ones, false ones and the number of non-detected anomalies).

5.9 First set of experiments

5.9.1 Patient 1

Patient 1 has four variables to monitor : Noninvasive Blood Pressure (NBP), Blood Temperature (BTemp), Arterial hemoglobin oxygen saturation (SpO2), Respiration Frequency (Resp) (see fig. 5.2). Each variable has its own Control Agent called respectively CtrlNBP, CtrlBTemp, CtrlSpO2, CtrlResp.

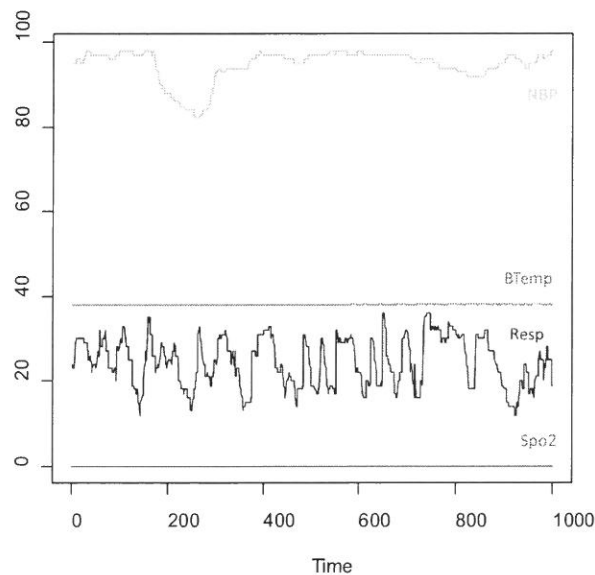


Figure 5.2: Subset of Patient 1 variables

As a first attempt, time series monitoring technique has been used as a procedure for the online detection of patterns in univariate time series and applied it to the observations of the four physiological variables seen above. Then each control agent identified and estimated the ARIMA model and so monitored around 42000 values. The results are presented in table 5.1.

	Model identified	Sensitivity	Specificity	Computing Time before Model Detection	Model Detection Computing Time	Computing Time after Model Detection
CtrlNBP	ARIMA(3,1,2)	100%	98%	5 to 6 ms	7 s 182 ms	if forecasting and model updating 50 to 60 ms else 5 to 6 ms
CtrlBTemp	ARIMA(3,1,3)	64%	60%	5 to 6 ms	6 s 737 ms	if forecasting and model updating 300 to 500 ms else 5 to 6 ms
CtrlSpO2	ARIMA(3,1,3)	98%	97.5%	5 to 6 ms	5 s 562 ms	if forecasting and model updating 300 to 500 ms else 5 to 6 ms
CtrlResp	ARIMA(1,1,1)	100%	98%	5 to 6 ms	7 s 771 ms	if forecasting and model updating 140 to 150 ms else 5 to 6 ms

Table 5.1: Patient 1 data monitored by Time Series based Control Agents

	Thresholds	Sensitivity	Specificity	Computing Time
CtrlNBP	90<NBP<220	94%	9,62%	5 to 6 ms
CtrlBTemp	37<BTemp<37,9	4%	10,72%	5 to 6 ms
CtrlSpO2	85<Spo2<100	94%	27%	5 to 6 ms
CtrlResp	5<Resp<25	47%	4,87%	5 to 6 ms

Table 5.2: Patient 1 data monitored by Threshold based Control Agents

The intuitive rule that we adopted for the detection of simple and patchy outliers, level changes and abrupt trends is to compare the incoming observation to the one-step ahead forecasting based on such model. This approach failed in tow cases :

1. In the presence of a gradual trend, when the first change point is not detected and so integrated into the updating of model parameters. In this case, the model will follow the trend and doesn't detect it. This deficiency implies that CtrlBTemp and CtrlSpO2 have respectively 64% and 98% sensitivity.
2. In the presence of a short outlier. If this outlier lies within the interval forecast, it will be integrated into the updating of the model parameters and so all the model will deviate and the normal values recorded after that will be considered as anomalies. This deficiency implies that all the Control Agents don't have a 100% specificity.

As a second attempt, a threshold based technique is used keeping the same limits mentioned in the annotation of the database (see table 5.2).

Because of the widening of the tolerance interval during the monitored period, the Control Agents were not able to detect all the anomalies, this may explain in some sense the performance of the Control Agent of not achieving a 100% sensitivity level.

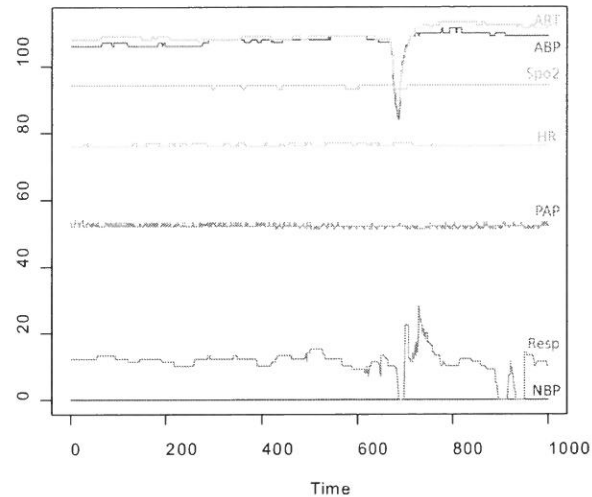


Figure 5.3: Subset of patient 2 variables' values

In this system, Control Agents, deal with a single value at a time and not with the usually changes of the variable over time. This would imply the how specificity levels performed.

5.9.2 Patient 2

Patient 2 has seven variables : Arterial Blood Pressure (ABP), Pulmonary Arterial Pressure (PAP), Heart Rate (HR), NBP, ART, Resp, SpO2 (see fig.5.3) . Each variable has its own Control Agent called respectively CtrlABP, CtrlPAP, CtrlHR, CtrlNBP, CtrlART, CtrlResp, CtrlSpO2.

The peculiarity of this database is the use of thresholds conform to safety standards. From the results in tables 5.3 and 5.4, it is easy to see that threshold based agents (CtrlPAP, CtrlART, CtrlSpO2) have a good sensitivity as well as the time series based agents. In contrast, time series based CtrlABP agent outperforms threshold based agent in terms of the sensitivity criterion. This could be explained by the fact that ARIMA models can detect abrupt changes even when these occur within the tolerance interval.

For variables such as NBP, HR and Resp no anomalies have been detected. Hence, no sensitivity and specificity computation was performed.

Time series based Control Agents are responsible for executing sequentially the following tasks:

1. Data collection,
2. Model identification and estimation, and one step ahead forecasting.

	Model identified	Sensitivity	Specificity	Computing Time before Model Detection	Model Detection Computing Time	Computing Time after Model Detection
CtrlABP	ARIMA(3,1,1)	97%	84%	5 to 6 ms	6 s 870 ms	if forecasting and model updating 140 to 250 ms else 5 to 6 ms
CtrlNBP	ARIMA(1,1,1)			5 to 6 ms	4 s 538 ms	if forecasting and model updating 70 to 90 ms else 5 to 6 ms
CtrlPAP	ARIMA(1,1,1)	100%	89,6%	5 to 6 ms	4 s 765 ms	if forecasting and model updating 80 to 100 ms else 5 to 6 ms
CtrlART	ARIMA(2,1,2)	100%	23%	5 to 6 ms	7 s 906 ms	if forecasting and model updating 150 to 300 ms else 5 to 6 ms
CtrlHR	ARIMA(2,1,2)			5 to 6 ms	6 s 35 ms	if forecasting and model updating 250 to 400 ms else 5 to 6 ms
CtrlResp	ARIMA(1,1,1)			5 to 6 ms	4 s 750 ms	if forecasting and model updating 20 to 30 ms else 5 to 6 ms
CtrlSpO2	ARIMA(3,1,3)	100%	13,5%	5 to 6 ms	8 s 675 ms	if forecasting and model updating 800 to 900 ms else 5 to 6 ms

Table 5.3: Patient 2 data monitored by Time Series based Control Agents

	Thresholds	Sensitivity	Specificity	Computing Time
CtrlABP	90<ABP<140	63,2%	29%	5 to 6 ms
CtrlNBP	60<NBP<120			5 to 6 ms
CtrlPAP	20<PAP<65	100%	0,1%	5 to 6 ms
CtrlART	60<ART<120	100%	37%	5 to 6 ms
CtrlHR	80<HR<150			5 to 6 ms
CtrlResp	5<Resp<25			5 to 6 ms
CtrlSpO2	89<Spo2<100	100%	1,5%	5 to 6 ms

Table 5.4: Patient 2 data monitored by Threshold based Control Agents

	Usually used system	Threshold based MAS	Time Series based MAS
Alarms generated number	29317	6563	1809
Significant generated Alarms number	504	504	769
Significant missed Alarms number	319	319	54
False Alarms number	28494	5740	1040
Alarms generated reduction	-	78%	94%
False alarms reduction	-	80%	96,4%
Sensitivity	61%	61%	93,5%
Specificity	1,7%	7,6%	45,5%
Alarm Computing Time	5 to 6 ms	7 ms	7 ms

Table 5.5: Systems outputs using Patient 1 data

3. Data reception and comparison with the forecast values:

if the received value lies with the interval forecast, it will be integrated into the updating of the model parameters and so the future one step ahead forecast value will be computed,

if the received value exceeds the interval forecast, the forecast value will be retained without neither updating nor new forecasting.

In tables 5.1 and 5.3 we present the agents computing time when they run on a Dual 2 Core CPU T5200 at 1,6GHz and 1GO RAM.

Despite the time that the Control Agent lost in the model identification, it makes up for it just after that when it waits for the future incoming values.

Threshold based Control Agents spend just the time of comparing incoming data to the limits and so the computing time is usually between 5 to 6 *ms* (see tables 5.2 and 5.4).

5.10 Second set of experiments

This section deals with assessing the value of the proposed alarm agent functionality. We test its ability to take into account the correlation structure over time of the variables being involved. The output of our system is compared with the output of the usually used system.

Considering correlations, threshold based multi-agent systems improve the specificity in comparison with the usually used system (see tables 5.5 and 5.6).

Introducing time dependant serial correlation and variability as depicted by our time series based multi-agent system resulted in a considerable gain as exemplified by the obtained specificity and sensibility ratio (see tables 5.5 and 5.6). This means that the number of false alarms decreased and the number of significant alarms increased.

The time spent to generate an alarm is between 5 to 6 *ms* for the usually used system. The time spent by the threshold based multi-agent system is the sum of the time spent by the Control Agent

	Usually used system	Threshold based MAS	Time Series based MAS
Alarms generated number	100382	39250	37444
Significant generated Alarms number	795	795	1019
Significant missed Alarms number	281	281	57
False Alarms number	99313	38181	36368
Alarms generated reduction	-	61%	63%
False alarms reduction	-	61,5%	63,4%
Sensitivity	74%	74%	94,7%
Specificity	0,8%	2%	3%
Alarm Agent Computing Time	5 to 6 ms	7 ms	7 ms

Table 5.6: Systems outputs using Patient 2 data

(5 to 6 *ms*) and the time spent by the Alarm Agent for the rule base consulting (7 *ms*). Likewise, the time spent by the time series multi-agent system is the sum of the time spent by the control agent in case of anomaly (5 to 6 *ms*) and the time spent by the Alarm Agent.

5.11 Conclusion

In conclusion, we can notice that considering physiological variable correlation and variability over the time has improved medical monitoring performance from sensitivity and specificity point of view without decreasing the online aspect.

Chapter 6

CONCLUSIONS AND FUTURE RESEARCH

6.1 Conclusions

In this work we presented ICU issues and medical staff concerns. We focused on the large number of variables that a physician has to deal with, and the monitoring system alarms which are numerous. Such alarms are produced by online monitoring systems based on fixed thresholds. This kind of systems produce alarms each time that a change point (as outlier, level change or trend) occurs and exceeds the threshold. In practice, we can see that:

1. A large number of false alarms; that are due to the presence of non-symptomatic outliers.
2. A large number of missed significant alarms; that are due to the width of the tolerance interval or to the non-detected abrupt changes that have occurred within the tolerance interval.

Some efforts were considered to improve ICU conditions. Several works tackled this problem but they didn't come to sizable conclusions. This alarming situation compelled us to suggest our online multi-agent monitoring system which tries to filter artifacts that cause false alarms in the first hand and produce significant ones in the second hand.

Because intensive care online monitoring data is generally high dimensional and multivariate, the more appropriate representation could be a multivariate time series approach. But, multivariate time series techniques have several limits; we cite as example the fact that they need a lot of memory and time execution. So, they are not appropriate to be used in online monitoring systems. Moreover, the peculiarity of multivariate time series is the presence of serial correlations. Hence, a change point means a change point at the same time in several variables (so called contemporaneous correlation) or a change point at different times (so called cross correlation). The detection of such points in multivariate time series is very difficult because there is a large number of directions for cross correlations.

To alleviate to this problem, we proposed to combine univariate time series technology which can be helpful to interpret data coming from the clinical information system, with multi-agent technology which can be helpful to consider contemporaneous and cross correlations and avoid dealing with multivariate time series.

Univariate time series analysis allows detecting change points which are important for assessing critical states. Usually the detection of a change point in a system leads to an alert, and so, the physician can check the situation and possibly take some therapeutic actions. A well known problem here is the occurrence of isolated and patchy outliers which make time series noisy and can produce false alerts. Hence there is a necessity for an automatic detection of outliers in online monitoring data.

For this purpose, we suggested to develop a new online multi-agent monitoring system. This system's principle is to automatically detect in real time, critical states efficiently from high dimensional data. We began by defining a new architecture. This architecture is based on a unified systemic approach. We have appointed that all components in the architecture at all levels have the same structure with 3 types of components:

1. a sensor,
2. a transformer,
3. and a deliver.

In this architecture some components are considered as software agents within specific behavior. The time series technology is incorporated in our architecture to handle the false and significant missed alarms problem.

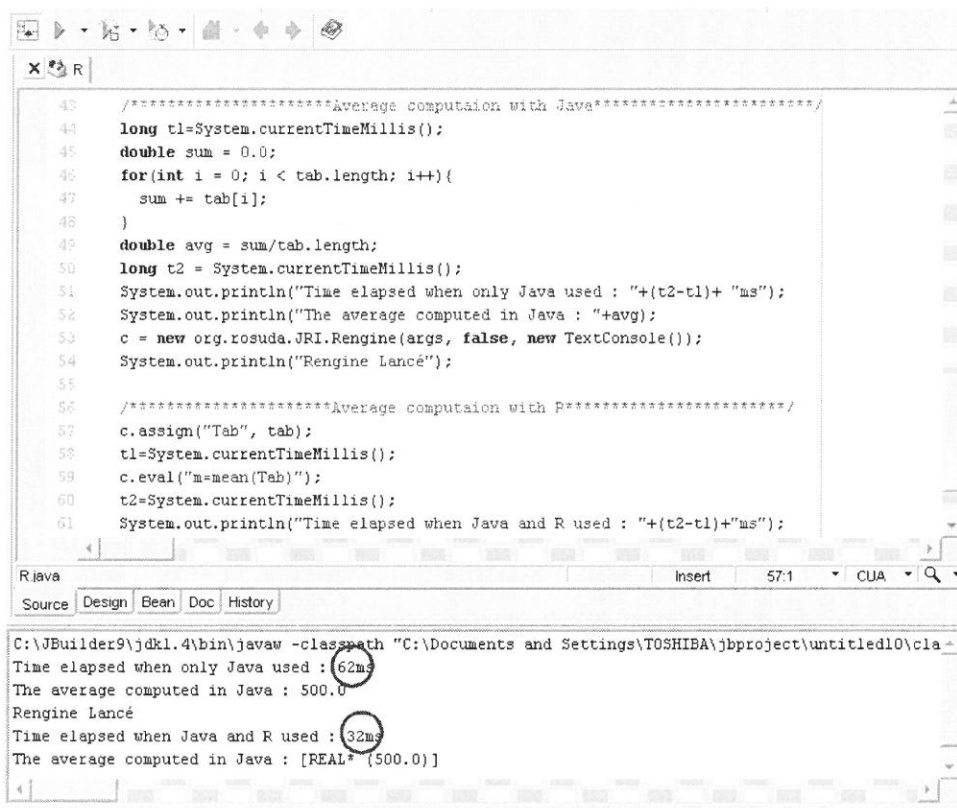
An ontology has been described and a general system structure with detailed explanation of the roles and interactions in accordance with AUML paradigm has been presented. The levels of the system architecture, logically and functionally connected, have been presented.

For the online recognition of anomalies, we identified and estimated ARIMA models during the equilibrium or steady patient's state and found the measure to detect deviations from the steady state. The nature and peculiarities of experimental data have been discussed.

In clinical practice, the physician typically select the most important variables (according to the experience) and bases his decision on the patterns found in these variables. In the presence of this kind of systems, the physician will not worry about the variables to be selected. He can select as many variables as he wants. In this case, even hidden anomalies present the other variables can be detected.

Because the system used in usual practice is not available, an effort has been made to redevelop it. We have also developed the threshold multi-agent system. Both of these two systems have been compared to our time series multi-agent system in terms of forecasting performance.

The previous three systems were developed in Java language, the multi-agent aspect is developed under the JADE (Java Agent DEvelopment) platform and the time series modules are developed in R language.



```

43  /*****Average computaion with Java*****/
44  long t1=System.currentTimeMillis();
45  double sum = 0.0;
46  for(int i = 0; i < tab.length; i++){
47      sum += tab[i];
48  }
49  double avg = sum/tab.length;
50  long t2 = System.currentTimeMillis();
51  System.out.println("Time elapsed when only Java used : "+(t2-t1)+ "ms");
52  System.out.println("The average computed in Java : "+avg);
53  c = new org.rosuda.JRI.Rengine(args, false, new TextConsole());
54  System.out.println("Rengine Lancé");
55
56  /*****Average computaion with R*****/
57  c.assign("Tab", tab);
58  t1=System.currentTimeMillis();
59  c.eval("m=mean(Tab)");
60  t2=System.currentTimeMillis();
61  System.out.println("Time elapsed when Java and R used : "+(t2-t1)+"ms");

```

C:\JBuilder9\jdk1.4\bin\javaw -classpath "C:\Documents and Settings\TOSHIBA\jbproject\untitled10\cla -
Time elapsed when only Java used : 62ms
The average computed in Java : 500.0
Rengine Lancé
Time elapsed when Java and R used : 32ms
The average computed in Java : [REAL* (500.0)]

Figure 6.1: Comparison between purely java program and a java program calling R method

R language is an Object Oriented language dealing with statistical programming. The peculiarity of this language is the fact that it is faster than Java in presence of statistical methods.

To support this choice, we developed two programs (see fig. 6.1) calculating the average of 10000000 values, the first program uses only Java language and the second calls R average method. We found that the second program took less time than the first.

To be able to access to R methods via Java programs, we used an interface called Java to R Interface (JRI).

In conclusion, we can say that ARIMA models have the ability to filter the outliers from the series, to detect change levels and abrupt trends. The weakness of this model is the graduate trends and short outliers.

6.2 Future research

From a theoretical point of view, a possible improvement of our system could be obtained by introducing other statistical modeling strategies based on adaptative algorithm for tracking trends and

pattern recognition. These methodologies have the potential of enhancing the capacity of the online monitoring system that we proposed by reducing the number of false alarms and improving its ability to adapt to new situation when new flows of data are entered to the system. In this case structural models and state space models could be considered as alternative to ARIMA specifications.

It is important to notice that the theoretical as well as the programming frameworks developed in this thesis could be easily adapted to implement the previously mentioned modeling strategies.

The proposed online monitoring system may be generalized and applied to other areas, such as economics, sociology, finance, environment, agriculture, etc. Some specific machine learning techniques, for example Support Vector Machine can be used in this kind of applications.

Appendix A

MAMS ANALYSIS AND DESIGN WITH AUML

There are many alternative agent-based system engineering methodologies including Agent UML (AUML) formalism [6], Prometheus [68], Multi-agent System Engineering methodology (MaSE) [25], among others. In this work, the analysis and design of the MAMS was made under AUML formalism.

The process of creating a multi-agent system in accordance with AUML, consists of two phases, which are :

1. **System Specification**, aimed to the identification of the system actors, goals, actions, scenarios and roles. In general, use case, activity, and class diagram extensions are used for this purpose.
2. **System analysis**, centered in the description and refinement of agent roles, general system structure and interaction protocols. Often sequence diagrams are used.

A.1 System Specification

A.1.1 Actors Model

MAMS agents communicate between them by messages using an ACL based protocol. The messages are sent to inform about the patient state or to request a specific treatment (see Fig. A.1).

A.1.2 Goals definition

The system is aimed to fulfil the enumeration of general goals. This enumeration includes a set of scenarios starting from physiological data reception and finishing with relevant monitoring patient (see Fig.A.2).

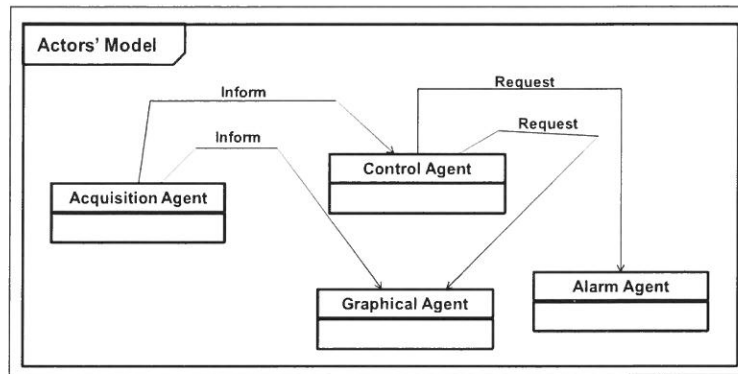


Figure A.1: MAMS Actors' Model

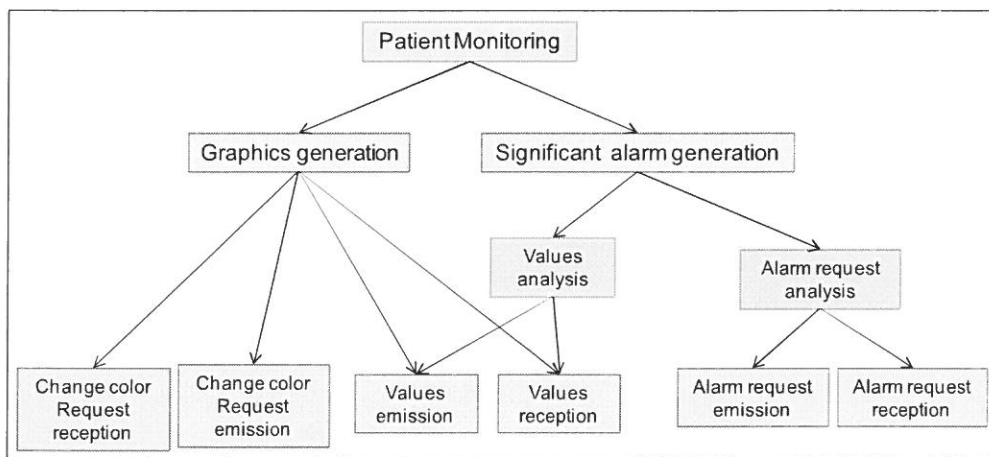


Figure A.2: The tree of goals

In our system there is a lot of secondary subgoals gathered from the main goal of the system which is the patient monitoring. The specification has led to the identification of the number of those goals and the actors in charge to make them :

1. **Acquisition System:** has as actors physiological variables to monitor. Those actors are reactive agents called Acquisition Agents and have as goals :
 - Control Agent **creation**.
 - Graphical Agent **creation**.
 - physiologic values **reception** from sensors linked to patient.
 - physiologic values **emission** to the monitoring system.
2. **Monitoring System:** has two kinds of actors created by the Acquisition Agents. Each Acquisition Agent creates its own Control Agent and Graphical Agent.

Control Agent is a cognitive actor and has as goals :

1.
 - Values **reception** from Acquisition Agent.
 - **Analysis** and anomaly detection.
 - Alarm request **emission** to Alarm System.
 - Change color request **emission** to the Graphical Agent in order to turn red the graph in the case of anomaly detection and to restore the original color when the anomaly goes.

The Graphical Agent is a reactive actor, having as goals :

1.
 - Values **reception** from Acquisition Agent.
 - Change color **reception** from Control Agent.
 - Graphics **display** in the data viewer.
3. **Alarm System:** has one cognitive actor which is the alarm agent. Its goals are :
 - Alarm request **reception** from Control Agents.
 - Alarm request **analysis**.
 - Alarm generation **decision**.

A.1.3 Roles definition

In addition to the goals, agents can perform various roles during their existence. Therefore, the implementation of an agent can satisfy different roles. Each Agent is identified by its goal and the ability that it needs to play its role. According to the system's goals we suggest the following roles: (1) Creator, (2) Emitter, (3) Receiver, (4) Analyzer, (5) Displayer, (6) Decision-maker.

The Acquisition Agent (see Fig. A.3) can be a receiver in order to receive values from sensor, an emitter to emit values to Control and graphical Agents or creator to create its corresponding Control and Graphical Agents.

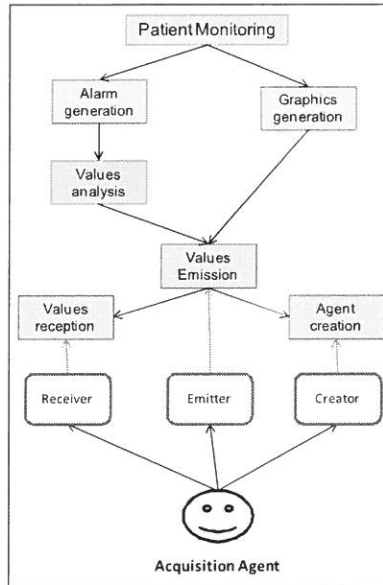


Figure A.3: Acquisition Agent goal diagram

The Control Agent (see Fig. A.4) can satisfy a receiver role to receive values from Acquisition Agent, an analyzer role to analyze data and an emitter role to emit Alarm and change color requests respectively to Alarm and Graphical Agents.

A Graphical Agent (see Fig. A.5) can be a receiver to receive values from Acquisition Agent and change color requests from Control Agent or a displayer to display generated graphics on the screen.

The Alarm Agent (see Fig. A.6) plays a receiver and a decision maker roles for the purposes of alarm request reception and alarm generation respectively.

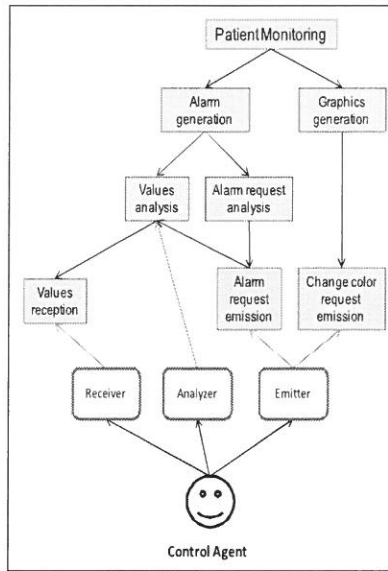


Figure A.4: Control Agent goal diagram

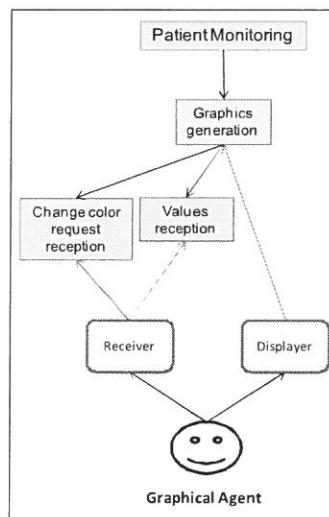


Figure A.5: The graphical Agent goal diagram

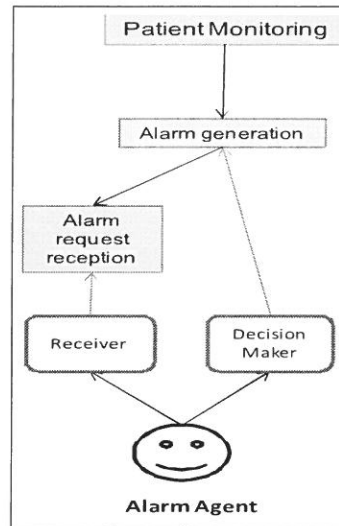


Figure A.6: Alarm Agent goal diagram

A.1.4 Use case model

Following the UML-based design approach, use case diagrams are applied to capture the interactions of the system with the user. In order to take into account the notion of agency, we distinguish between ordinary users and agents when they are modeled as actors. The following models will clarify how the system interacts with external actors in the first hand, and the intern agents interacts with the various sub-systems, in the second hand.

A.1.5 MAMS use case diagram

Figure A.7 illustrates the general MAMS use case diagram. In this diagram we present three use cases that a medical staff can do on the system. In fact, the medical staff as external actor, composed of physicians and nurses, acts on the system either to activate and deactivate it or to monitor patient's state. Each use case can in turn be detailed as follows.

Figure A.8 illustrates the system activation use case diagram. In this diagram we explain the sequence of events to activate the system. When the patient is connected to the system via the sensors, an agent called Starting agent is created having as goal to create the Acquisition Agents and the Alarm Agent. The Acquisition Agents will act in the Acquisition System and the Alarm Agent will act in the Alarm System.

Figure A.9 presents the different use cases done by each Acquisition Agent present in the Acquisition System. And Figure A.10 illustrates the interaction between the monitoring System and the

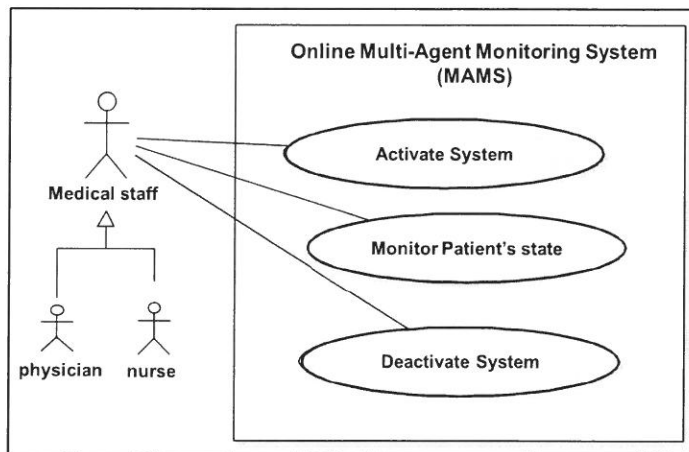


Figure A.7: MAMS use case diagram

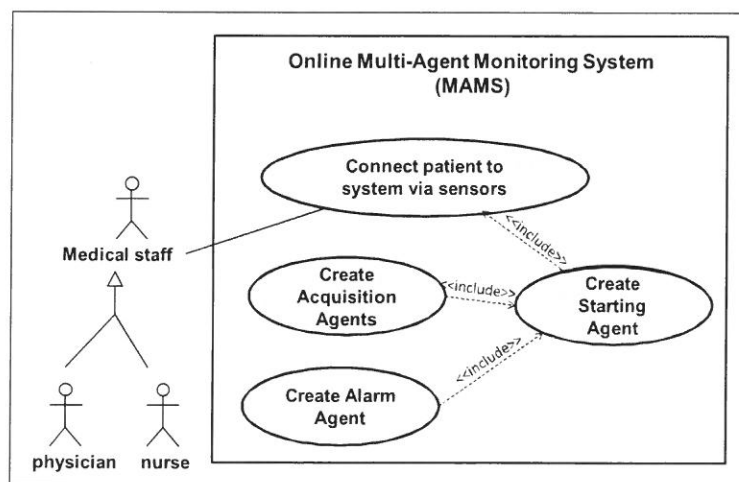


Figure A.8: System activation use case diagram

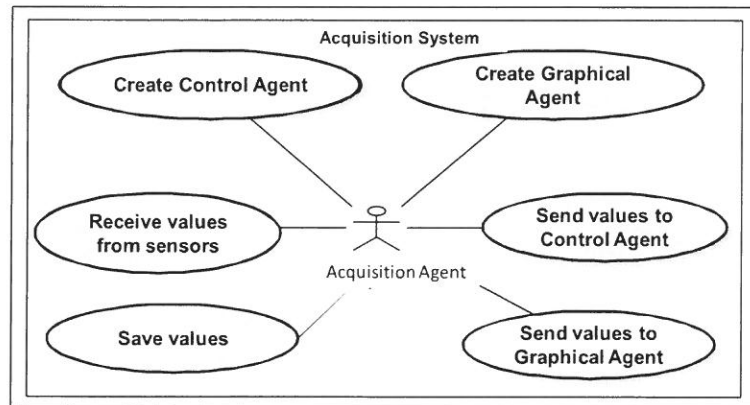


Figure A.9: Acquisition System Use Case Diagram

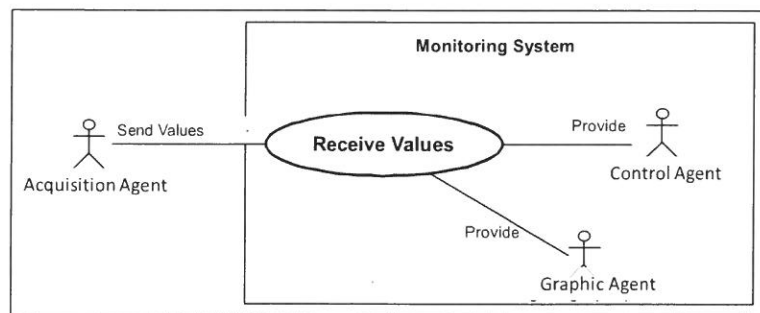


Figure A.10: Acquisition Agents and monitoring system interaction use case diagram

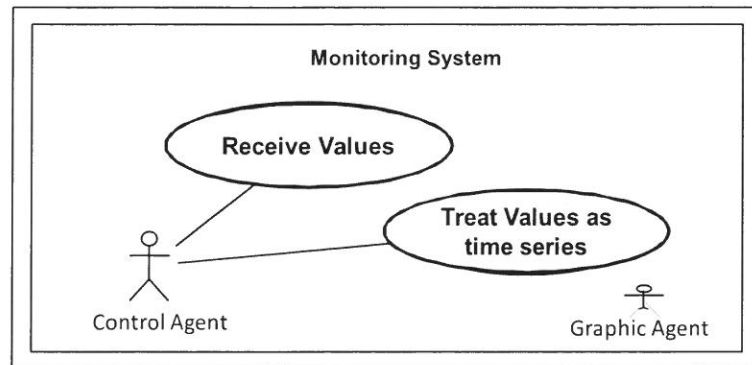


Figure A.11: Control Agent use case diagram

Acquisition agents as external actors. The interaction is represented by an ACL message, containing the received value, sent by the Acquisition Agent to its corresponding Control and Graphical Agents.

In the monitoring System, Control Agents receive values from Acquisition Agents and treat them using monitoring techniques (see Fig. A.11). In case of anomaly detection as seen in Fig. A.12, an alarm request is sent to the Alarm Agent and a change color request is sent to the corresponding Graphical Agent in order to turn red the graphic color. And in case of normal state, an alarm cancel request (RAZ) is sent to the Alarm agent and a change color request to the corresponding Graphical Agent in order to restore the original color.

In Figure A.13 we see the interaction using ACL messages between Monitoring System intern actors such as Control and Graphical Agents. And Fig. A.14 represents the graphical Agent behavior towards the ACL messages received from Acquisition and Control Agents.

The Control Agent interacts with the Alarm System using ACL message in the aim of sending an alarm request (see Fig. A.15) or a RAZ request (see Fig. A.16).

The sensor disconnecting, as seen in Fig.A.17, causes the Acquisition Agent destruction. For each Acquisition Agent destroyed the corresponding Control and graphical agents will be destroyed. If all the Control Agents are destroyed, the alarm Agent will be destroyed in turn.

The global use case in Fig. A.18 presents the different kinds of ACL messages circulated through the different layers of the system.

We will end the system specification by an activity diagram presented in Fig. A.19. The diagram shows the operational step by step workflow of several agents of our system (Starting agent, Acquisition Agent, Control Agent, Graphical Agent and alarm Agent).

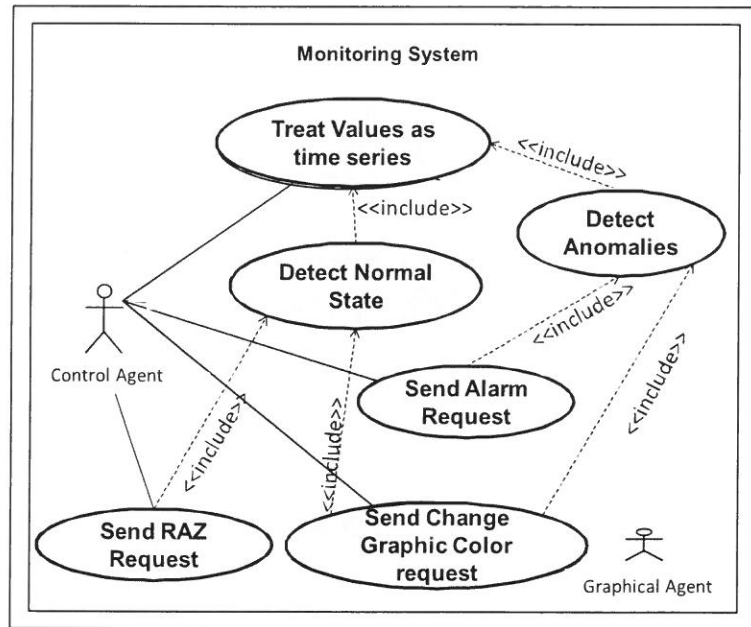


Figure A.12: detailed Control Agent use case

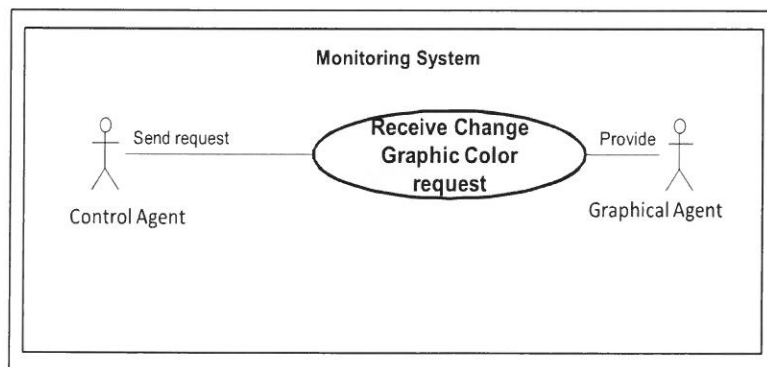


Figure A.13: Control and Graphical Agents interaction use case diagram

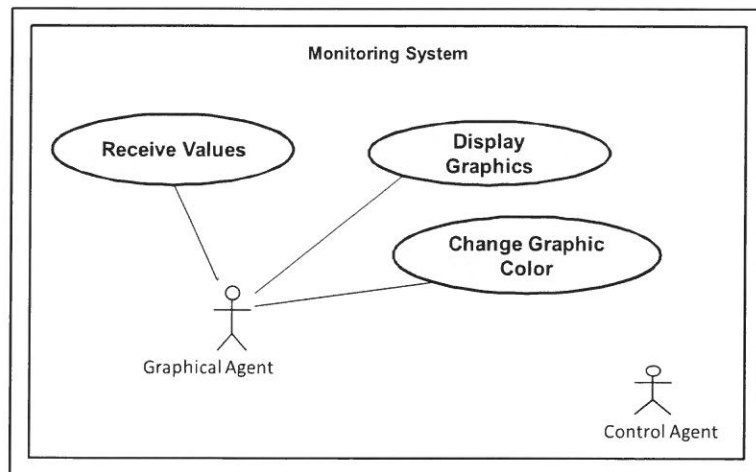


Figure A.14: Graphical Agent use case diagram

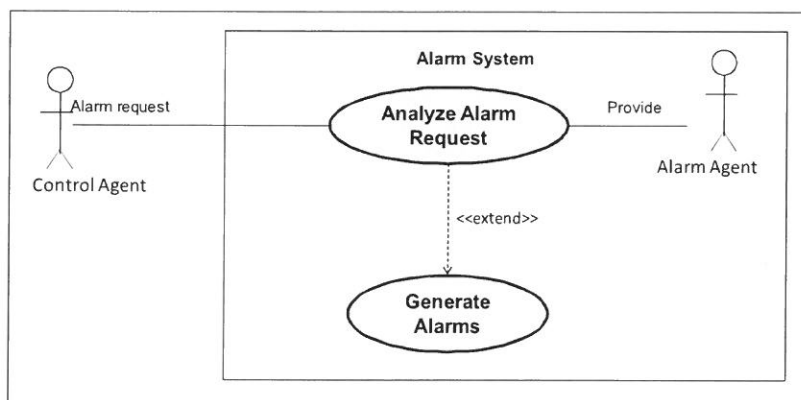


Figure A.15: Intraction between Alarm system and Control Agents as extern actors (a)

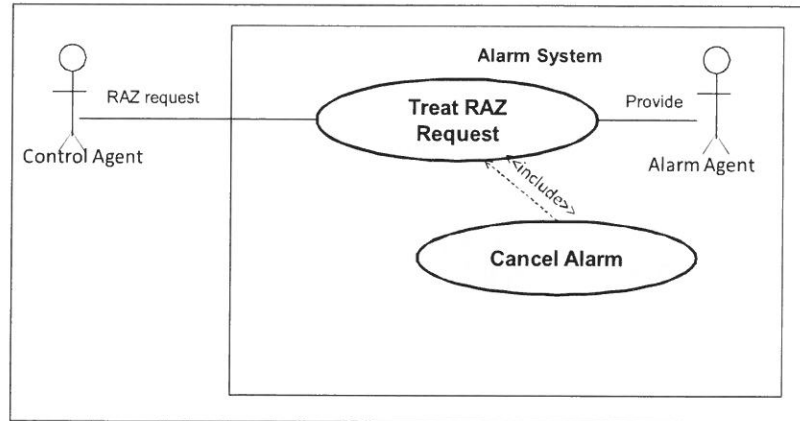


Figure A.16: Interaction between Alarm system and Control Agents as extern actors (b)

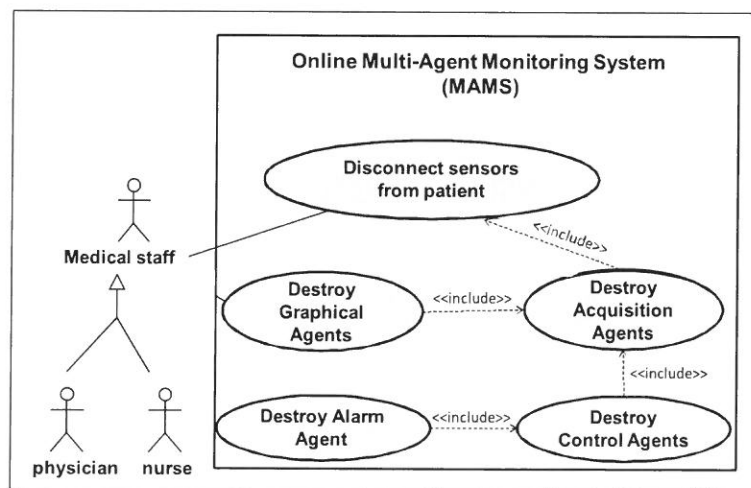


Figure A.17: MAMS turn off use case diagram

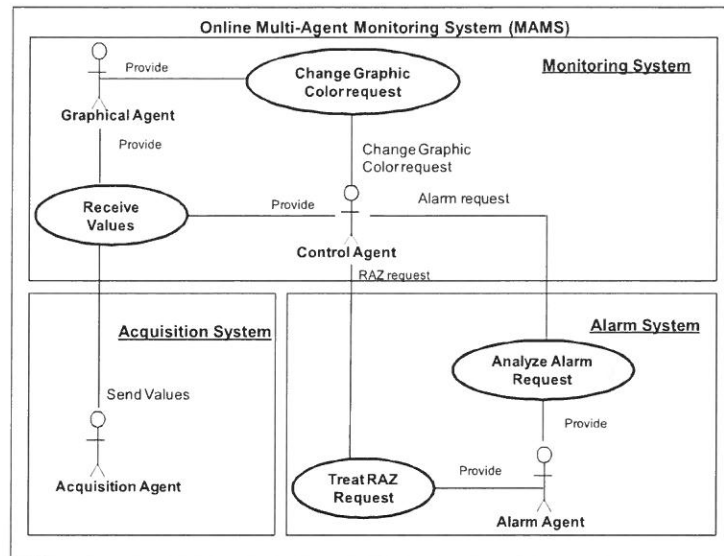


Figure A.18: Global use case diagram and the interaction between agents

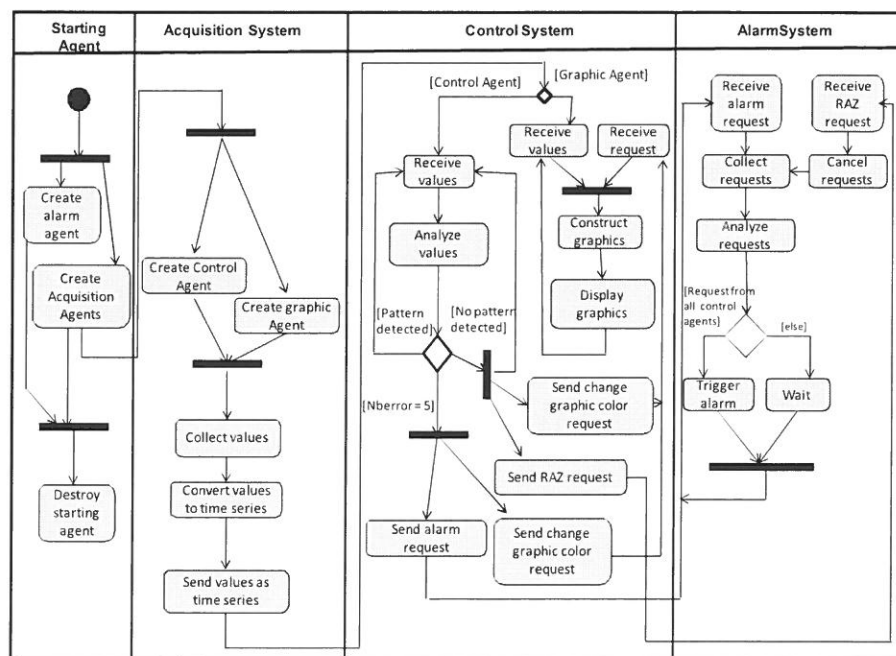


Figure A.19: MAMS global activity diagram

When turn on, the system will instantiate an agent from the Starting System Class. This agent is called the Starting Agent. Its aim is to create the necessary Acquisition agents and the Alarm Agent. When finished it destroys itself. Each Acquisition Agent creates in turn its corresponding Control and Graphical Agents in the aim of sending to them an ACL message containing values received from the sensor in a regular time interval. The Control Agent models the collected values using monitoring techniques, detects the anomaly. When five (5) consecutive values are detected, the Control Agent decides that the patient presents an anomaly and sends two ACL messages, one to the Graphical Agent in order to redden the relative graphic and the second to the Alarm Agent demanding an alarm generation. The Alarm Agent decides in function of the collected requests. The Alarm Agent bases its decision on a rule base presenting the interdependence of the physiological variables. When the requests coming from a list of interdependent variables the Alarm Agent generates an alarm.

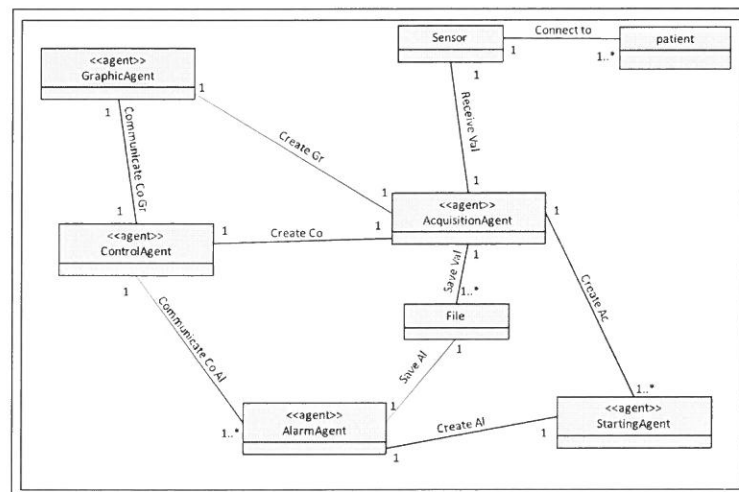


Figure A.20: Global class diagram

Fig. A.20 shows the different classes of the system and their relationships.

A.2 System Analysis

In the analysis phase we introduce the sequence diagrams which depicts the interaction between the different roles of the agents.

The Acquisition Agent has an initial role to create ontrol and Graphical agents (see fig. A.21). After that, it becomes responsible for data trasmission every m milliseconds (m will be defined by the staff in function of the disease) to the Control and Graphical Agents (see fig. A.22).

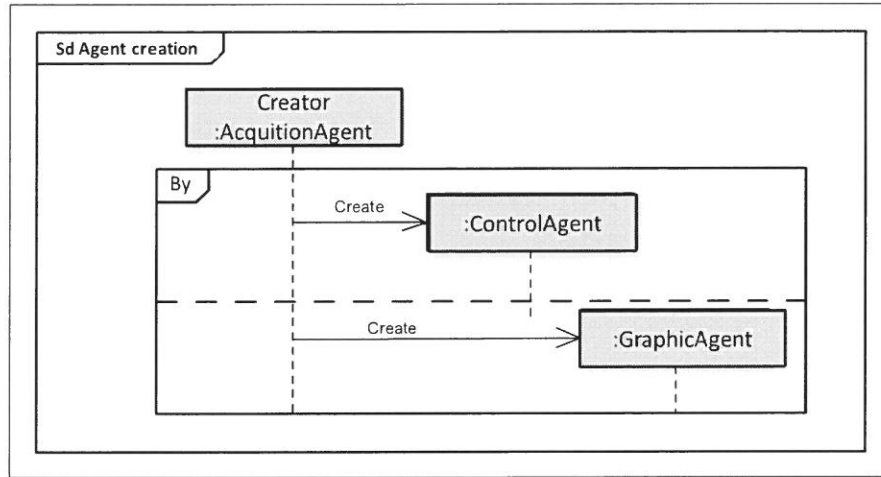


Figure A.21: Agent creation sequence diagram

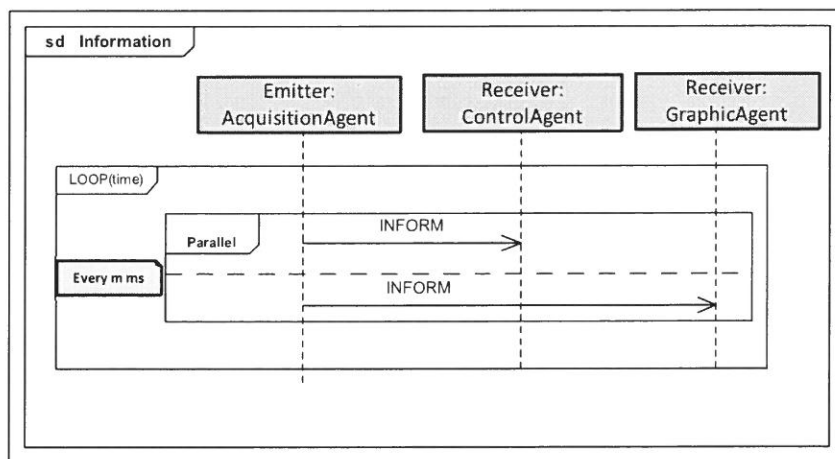


Figure A.22: Information reception from Acquisition Agent sequence diagram

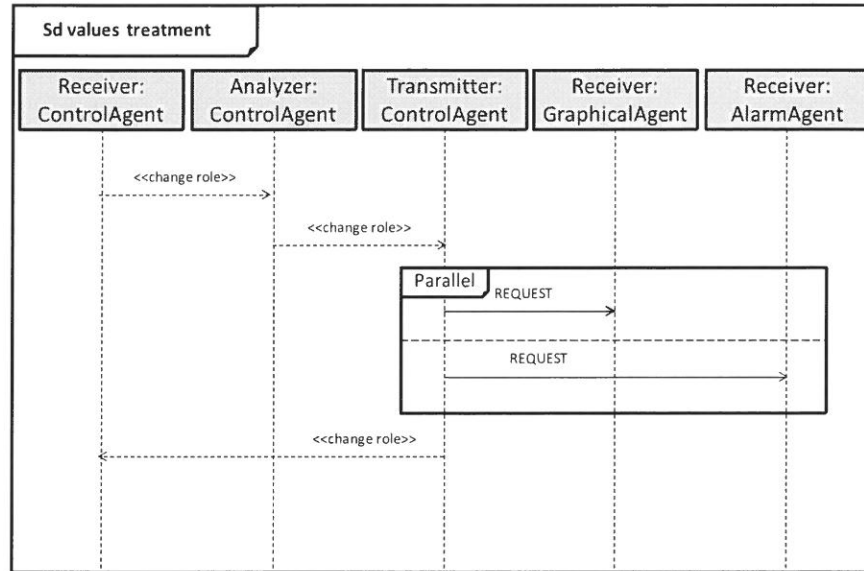


Figure A.23: Values treatment sequence diagram

Control Agent plays, in turn, several roles. Initially it is receiver, it receives data from Acquisition Agent. Immediately after that, it becomes analyzer, it analysis data. In case of anomaly or *RAZ* it changes role to transmitter and sends in parallel requests to Graphical and Alarm Agents (see fig. A.23). After that, it restores its initial role.

The Graphical Agent as receiver, receives requests from Control Agent changes its role to displayer in order to change graphic color (see fig.A.24) and restores its initial role. As receiver, the Alarm Agent receives requests from the Control Agent, changes role to Decision Maker and so decides to trigger or simply save alarm information. After that, it restores its initial role waiting for another request (see fig.A.25).In case of *RAZ*, the Control Agent as transmitter, sends request to both Graphical and Alarm Agents. The Graphical Agent follows the "change graphic color request" sequence diagram of fig.A.24). The Alarm Agent as receiver, receives the request, changes its role and so cancels the Alarm (see fig.A.26).

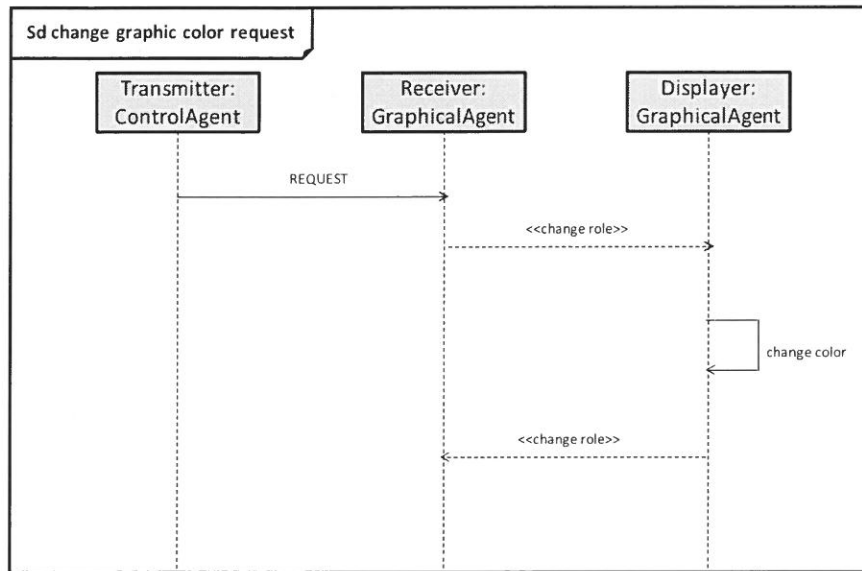


Figure A.24: Change graphic color sequence diagram

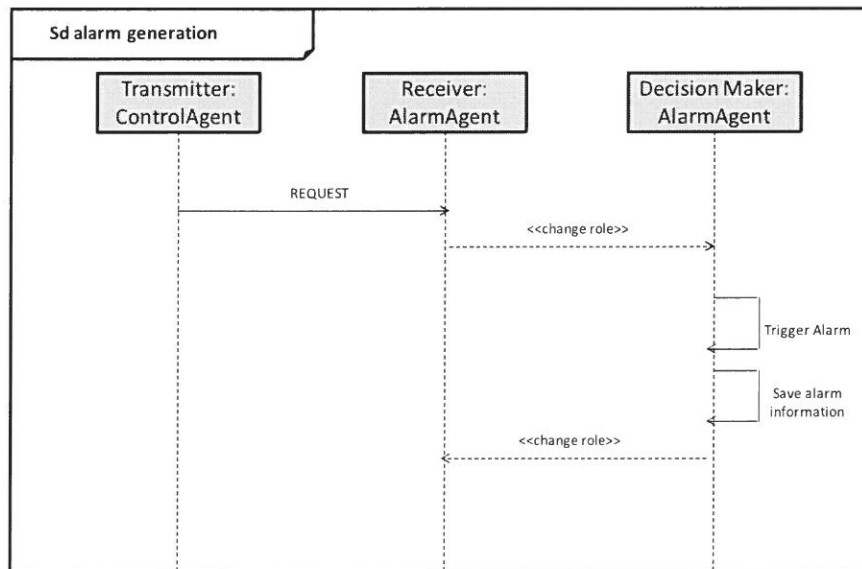


Figure A.25: Alarm generation sequence diagram

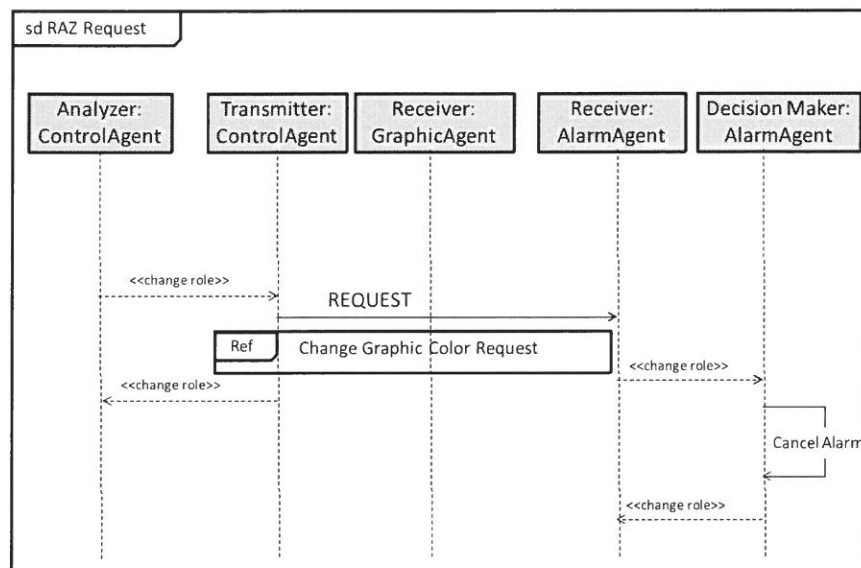


Figure A.26: RAZ request sequence diagram

Bibliography

- [1] Agazzi, E. (2001) , From technique to technology : the role of modern science, in H. Lenk and M. Maring (eds), advances and problems in the philosophy of technology, Münster/Hamburg/London, LIT verlag, pp 121-128.
- [2] Bach, F., R. and Jordan, M, (2004), Learning Graphical Models for Stationary Time Series, IEEE transactions on signal processing 52 (8) pp.2189-2199.
- [3] Baragona R.and Calzini C., (1999), Genetic algorithms for outlier identification of additive and innovational type in time series, Technical report, 11/1999, Universita degli Studi di ROMA "La Sapienza".
- [4] Bauer, M., Gather, U. and Imhoff, M. (1998), Analysis of high dimensional data from intensive care medicine. In Payne, R., editor, Proceedings in computational statistics, Springer Verlag, Berlin, pp. 185-190.
- [5] Bauer, M., Gather, U. and Imhoff, M. (1999), The identification of multiple outliers in online monitoring data, Technical report, SFB475, Department of statistics, university of Dortmund, Germany.
- [6] Bauer B., Müller J.P., and Odell J., (2001), Agent UML : a formalism for specifying multiagent software systems, International Journal of Software Engineering and Knowledge Engineering, 11(3), pp.207-230.
- [7] Becker, C. and Fried, R., (2002), Sliced Inverse Regression for high-dimensional time series, In O. Opitz, M. Schwaiger (Hrsg.): Exploratory Data Analysis in Empirical Research. Proceedings of the 25th Annual Conference of the German Classification Society, pp.3-12.
- [8] Becker C. and Gather U., (1998), The maximum asymptotic bias of outlier identifiers, Technical report, 3/98, SFB475, Department of statistics, university of Dortmund, Germany.
- [9] Becker C. and Gather U., (1999), The masking breakdown point of multivariate outlier identification rules, Journal of American Statistics Association, 94, pp.947-955.
- [10] Becker C. and Gather U., (2001), The largest non identifiable outlier : a comparison of multivariate simultaneous outlier identification rules, Computational Statistics & Data Analysis, 36, pp.119 - 127.

-
- [11] Bertalanffy, V., L., (1968). General System Theory. Foundations, Development, Applications. George Braziller Incorporated, New York.
- [12] Berwick DM. and Leape L L., (2006), Perfect Is Possible Pilot projects at hundreds of hospitals around the country prove that medical error rates can be reduced to zero, Newsweek, 16 October 2006, <http://www.msnbc.msn.com/id/15173079/site/newsweek/> (april 2007).
- [13] Box G. and Jenkins G., (1969), Time series analysis: Forecasting and control, San Francisco: Holden-Day.
- [14] Brillinger D. R., (1981), Time series. Data analysis and Theory, San Francisco, Holden Day.
- [15] Cantwell R.H. and Scevak J., (2004), Discrepancies between the “ideal” and “passable” doctorate: Supervisor thinking on doctoral standards, Refereed paper presented at the Annual Conference of the Australian Association for Research in Education, Melbourne.
- [16] Carroll R.J. and Li K.C., (1995), Binary regressors in dimension reduction models : a new look at treatment comparisons, *Statistica, Sinica* 5, pp.667-688.
- [17] Chambrin MC., (2001), Alarms in the intensive care unit: how can the number of false alarms be reduced?, *Critical Care*, 5, pp.184-188.
- [18] Chebrolu S., Abraham A. and Thomas J.P., (2005), Feature deduction and ensemble design of intrusion detection systems, *Computer & Security*, 24, pp. 295-307.
- [19] Clancy C M., (2006), The Intensive Care Unit, Patient Safety, and the Agency for Healthcare Research and Quality, *American Journal of Medical Quality*, 21(5), pp. 348-351.
- [20] Dahlhaus, R., (2000), Graphical interaction models for multivariate time series, *Metrika* 51, pp.157-172.
- [21] Daumer M., Bauer M. and Lederer C., (1997), Average run length and mean delay for changepoint detection : robust estimates for threshold alarms, Discussion Paper 95, SFB 386.
- [22] Daumer M. and Falk M., (1998), On-line detection for state space models using multi-process Kalman Filters, in proceedings of the international linear algebra society symposium on fast algorithms for control, signals and image processing, Ed. D. O’Leary, Amsterdam, Elsevier, pp.125-135.
- [23] Daumer M., Falk M. and Beyer U., (1996), On-line monitoring using Multiprocess Kalman Filtering, Discussion Paper 54 of SFB 386.
- [24] Davies P.L., Fried R. and Gather U., (2002), Robust Signal extraction for on-line monitoring data. Technical report 1/ 2002, SFB 475, University of Dortmund, Germany.
-

-
- [25] DeLoach S. A., Wood M. F. and Sparkman C. H., (2001), Multiagent systems engineering. *International Journal of Software Engineering and Knowledge Engineering*, 11(3), pp. 231-258.
- [26] Donchin Y, Gopher D, Olin M., Badihi Y., Biesky M., Sprung C.L., Pizov R. and Cotev S., (1995), A look into the nature and causes of human errors in the intensive care unit, *Critical care medicine*, 23(2), pp. 294-300.
- [27] Donchin Y. and Seagull F.J., (2002), The hostile environment of the intensive care unit, *Current Opinon in Critical Care*, 8, pp. 316-320.
- [28] Dragoslav D. S., (1991), *Decentralized control of complex systems*, Academic Press Inc., New York.
- [29] Fineman B., (2004), *MetaMonitor: a system for patient monitoring in intensive care units*, Master Thesis, Carnergie Mellon University, www.mildabandon.com (april 2007).
- [30] Flaatten H., Hevroy O., (1999), Errors in the intensive care unit (ICU) : Experiences with an anonymous registration, *Acta anaesthesiologica scandinavica (Acta anaesthesiol.scand.)*, 43 (6), pp. 614-617.
- [31] Fried R. and Gather U., (2002), Fast and robust filtering of time series with trends, *COMPSTAT 2002 Proceedings in Computational Statistics*, Springer, Berlin, pp.367-372.
- [32] Fried, R., Gather, U. and Imhoff, M. (2002), Pattern recognition in intensive care online monitoring, In M. Schmitt et al. (Hrsg.): *Computational Intelligence Techniques in Medical Diagnosis and Prognosis*, Springer Verlag, pp.129-165.
- [33] Fried, R., Gather, U., Imhoff, M. and Bauer, M. (2000), *Statistical Methods in Intensive Care Online Monitoring*, Medical Data Analysis, Springer, Berlin, pp.67-77.
- [34] Fried R., Gather U., Imhoff M., Keller M. and Lanius V., (2002), Combining Graphical models and PCA for statistical process control, *Conference COMPSTAT 2002*, Berlin, Germany.
- [35] Gather U., (1999), Robust Dimension Reduction, *Bulletin of the International Statistical Institute, Proceedings Acts*, 52nd Session, Tome LVIII, Finland.
- [36] Gather U., Fried R. and Imhoff M., (2000), On-line classification of states in intensive care, In W. Gaul, O. Opitz, M. Schader (Hrsg.): *Data Analysis. Scientific Modeling and Practical Application*. Springer, Berlin, pp.413-428.
- [37] Gather U., Fried R., Lanius V. and Imhoff M., (2001), On-line monitoring of high dimensional physiological Time series - a case study, *Estadistica*, 53, pp.259-298.
- [38] Gather U., Hilker T. and Becker C., (1998), *Robust Sliced Inverse Regression Procedures*, Technical report 22/98, SFB 475, University of Dortmund, 44221 Dortmund, Germany.
-

-
- [39] Gather U. and Imhoff M., (2002), Graphical models for multivariate time series from intensive care monitoring. *Statistics in Medicine*, 21, pp.2685-2701.
- [40] Goldberger A.L., Amaral L.A.N., Glass L., Hausdorff J.M., Ivanov P.Ch., Mark R.G., Mietus J.E., Moody G.B., Peng C.K. and Stanley H.E., (2000), PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* 101(23), pp.e215-e220 [<http://circ.ahajournals.org/cgi/content/full/101/23/e215>], (June 13 2000).
- [41] Heravi S., Osborn D.R. and Birchenhall C.R., (2004), Linear versus neural network forecasts for European industrial production series, *International Journal of Forecasting* 20, pp.435-446.
- [42] Hudson L., (1985), Monitoring of critically ill patients, conference summary. *Respiratory Care*, 30, pp.628-636.
- [43] IMD-Soft Inc., (2007), MetaVisionTM-The paperless ICU system. www.imd-soft.com (december 2009).
- [44] Imhoff M. and Bauer M., (1996), Time series analysis in critical care monitoring, *New Horizons* 4 (4), pp.519-513.
- [45] Imhoff M. , Bauer M., Gather U. and Lohlein, D., (1997), Time series analysis in intensive care medicine. *Applied Cardiopulmonary Pathophysiology* 6, pp.263-281.
- [46] Imhoff M., Bauer M., Gather U. and Lohlein D., (1998), Statistical pattern detection in univariate time series of intensive care on-line monitoring data, *Intensive Care Medicine*, 24, 1305-1314.
- [47] Imhoff M. and Kuhls S., (2006), Alarm Algorithms in Critical Care Monitoring, *Anesth Analgesia*, 102, pp 1525-1537.
- [48] Jenning D., Amabile T. and Ross L., (1982), Informal covariation assessments : data-based versus theory-based judgements, *Judgment Under Uncertainty : Heuristics and biases*, Cambridge : Cambridge University Press, pp.211-230
- [49] Justel A., Pena D., Tsay R.S., (2001), Detection of outlier patches in autoregressive time series, *Statistica Sinica* 11, pp.651-673.
- [50] Kim H.S., Cha S.D., (2005), Empirical evaluation of SVM-based masquerade detection using UNIX commands, *Computer & Security*, 24, pp.160-168.
- [51] Krystal A.D., Prado R. and West M., (1999), New methods of time series analysis of non-stationary EEG data : Eigenstructure decompositions of time varying autoregression, *Clinical Neurophysiology*, 110, pp.2197-2206.

-
- [52] Landrigan C. P., Rothschild J. M., Cronin J. W., Kaushal R., Burdick E., Katz J. T., Lilly C. M., Stone P. H., Lockley S. W., Bates D. W. and Czeisler C. A., (2004), Effect of Reducing Interns' Work Hours on Serious Medical Errors in Intensive Care Units, *The New England Journal of Medicine*, 351 (18), pp.1838-1848.
- [53] Li K.C., (1991), Sliced Inverse Regression for dimension reduction (with discussion), *Journal of the American Statistical Association*, 86, pp.316-342.
- [54] Liu X., Tucker A., Cheng G. and Loizou G., (1999), Modelling Multivariate Time Series, *Intelligent Data Analysis in Medicine and Pharmacology* pp.67-78.
- [55] McCulloch R.E. and Tsay R.S., (1994), Bayesian analysis of autoregressive time series via the Gibbs sampling. *Journal of Time Series Analysis*, 15, pp.235-250.
- [56] Marcellino M., (2005), Instability and non-linearity in the EMU. In: C. Milas, P. Rothman and D. van Dijk, Editors, *Nonlinear Time Series Analysis of Business Cycles*, Elsevier, Amsterdam.
- [57] Miller G. (1956), The marginal number seven, plus or minus two : some limits to our capacity for processing information, *Psychological Review*, 63, pp.81-97
- [58] Morik K., Imhoff M., Brockhausen P., Joachims T. and Gather U., (2000), Knowledge Discovery and Knowledge Validation in Intensive Care, *Artificial Intelligence in Medicine*, 19(3), pp.225-49.
- [59] Morris A.H., Chapman R.H. and Gardner R.M., (1985), Frequency of wedge pressure errors in the ICU, *Critical Care in Medicine*, 13(9), pp.705-708.
- [60] Noura K. and Trabelsi A., (2006), Temporal Multi-Agent System in Intensive care Monitoring, Poster in International Symposium on Computational Biology & Bioinformatics, Bhubaneswar, India.
- [61] Noura K. and Trabelsi A., (2006), Time Series Analysis and Outlier Detection in Intensive Care Data, In proceedings of the 8th International Conference on Signal Processing (ICSP'06). IEEE, 2006, ISBN: 0-7803-9736-3, vol 4, pp 2499-2502.
- [62] Noura K., Ferchichi A. and Trabelsi A., (2006), Architecture of Temporal Online Monitoring Multi-Agent System in Intensive Care Units, In the 7th Tunisia-Japan Symposium on Society, Science & Technology (TJSSST 2006), Sousse, Tunisia.
- [63] Noura K., Ferchichi A. and Trabelsi A., (2007), Temporal Multi-Agent Based Architecture for Medical Monitoring Systems in Intensive Care Units: a Unified Systemic Approach, In proceedings of the Septièmes Journées Scientifiques des Jeunes Chercheurs en Génie Electrique et Informatique (GEI'2007), CPU, pp 109-118.
-

-
- [64] Noura K., Ferchichi A. and Trabelsi A., (2007), "Temporal multi-agent system in intensive care monitoring". In proceedings of the 24th Annual Conference and Exhibition HEALTHCARE COMPUTING (HC2007). British Computer Society (BCS), pp.38-46.
- [65] Noura K., and Trabelsi A., (2007),"Time Series technology in pattern recognition: case study in medical support systems". In proceedings of the 4th International Multi-Conference on Systems, Signals & Devices (SSD-07). IEEE. Conference on Communication and Signal Processing(CSP), volume 3.
- [66] Noura K. and Trabelsi A., (2010), Online Multiagent Monitoring in Intensive Care Units, to appear in proceedings of International Colloquium on Computing, Communication, Control, and Management(CCCM 2010), IEEE Catalog Number: CFP1048E-PRT; ISBN: 978-4244-7380-9, August 20-22, 2010, Yangzhou China.
- [67] Oates, T., Schmill M.and Cohen P., (1999), Efficient mining of statistical dependencies. Proceedings of the 16th International joint conference on artificial intelligence.
- [68] Padgham L., and Winikoff M., (2004), Developing intelligent agent systems: a practical guide. John Wiley and sons, RMIT University, Melbourne, Australia.
- [69] Prado R., Huerta G. and West M., (2000), Bayesian time-varying autoregressions : Theory, methods and applications, Special issue on Time Series and Related Topics of "Resenhas", Journal of the Institute of Mathematics and Statistics of the University of Sao Paolo, 4, pp.405-422.
- [70] Prado R. and West M., (1997), Exploratory modelling of multiple non-stationary time series : latent process structure and decompositions, In Modelling Longitudinal and Spatially Correlated Data (ed: T. Gregoire), Springer-Verlag.
- [71] Prielipp R C., Lewis K. and Morell R. C.,(1998),Ventilator Failure in the ICU: Déjà Vu All Over Again, Anesthesia Patient Safety Foundation Newsletter,13 (3), http://www.apsf.org/resource_center/newsletter/1998/fall/09vent.html (april 2007).
- [72] Ruping S., (2001), SVM Kernels for Time series analysis, Technical report 43/01, SFB 475, University of Dortmund, 44221 Dortmund, Germany.
- [73] Rech G., (2002), Forecasting with artificial neural network models, SSE/EFI Working Paper Series in Economics and Finance vol. 491, Stockholm School of Economics.
- [74] Schoenberg R., Sands D.Z. and Safran C., (1999), Making ICU alarms meaningful: a comparison of traditional vs. trend-based algorithms, Proceedings AMIA Symposium', pp. 379-83.
- [75] Stock J.H. and Watson M.W., (1999), A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series. In: R.F. Engle and H. White, Editors, Cointegration,
-

-
- causality and forecasting. A festschrift in honour of Clive W.J. Granger, Oxford University Press, Oxford, pp. 1–44.
- [76] Smith A.F.M. and West M., (1983), Monitoring renal transplants : an application of the multiprocess Kalman Filter. *Biometrics* 39, pp.867-878.
- [77] Sokolova M. and Fernandez-Caballero A., (2009), Modeling and implementing an agent-based environmental health impact decision support system, *Expert Systems with Applications*, 36, pp.2603-2614.
- [78] Taplin R. H., (1996), Model selection for time series in the presence of outliers, *Proceedings of Computing Science and Statistics: 28th Symposium on the Interface*, Sydney, Australia.
- [79] Terävirta T., Dijk D. V. and Medeiros M. C., (2005), Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series : A re-examination, *International Journal of Forecasting*, 21 (4), pp755-774.
- [80] Tkacz G., (2001), Neural network forecasting of Canadian GDP growth, *International Journal of Forecasting*, 17 (1), pp 57-69.
- [81] Tsien C. L., (1997), Reducing false alarms in Intensive Care Units : A systematic comparison of four algorithms, *Annual Fall Symposium*.
- [82] Tsien C. L., (1997), Poor prognosis for existing monitors in the intensive care units [Clinical Investigation], *Critical Care Medicine*, 25, pp.614-619.
- [83] Tsien C. L., (2001), TrendFinder: Automated Detection of Alarmable Trends, PhD Thesis in Electrical Engineering and Computer Science, Massachusetts Institute of Technology.
- [84] Tukey J.W., (1977), *Exploratory data analysis*, Princeton University, Addison-Wesley, 688 pages.
- [85] Wei W., (1990), *Time series analysis, Univariate and multivariate methods*, Addison-Wesley Publishing Company.
- [86] West M., (1997), Bayesian time series : Models and computations for the analysis of time series in the physical sciences, In *Maximum Entropy and Bayesian Methods*, 15, pp. 23-34.
- [87] Ye Z. and Weiss R.E., (2000), *Classes of dimension reduction methods*.
- [88] Zhang G., Patuwo B. E. and Hu M.Y., (1998), Forecasting with artificial neural networks: The state of the art, *International Journal of Forecasting*, 14 (1), pp 35-62.
- [89] Zhang Y., (2003), *Real-Time Analysis of Physiological Data and Development of Alarm Algorithms for Patient Monitoring in the Intensive Care Unit*, Master thesis of Engineering in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology.
-