Smart Monitoring Systems

for Alert Generation During Anaesthesia

Mirza Mansoor Baig

A thesis submitted to

Auckland University of Technology

in partial fulfilment of the requirements for the degree

of

Master of Engineering (ME)

March 2010

School of Engineering

Primary Supervisor: Hamid GholamHosseini

Acknowledgements

In the first place, I would like to record my gratitude to Dr. Hamid GholamHossieni for his supervision, advice, and guidance from the very early stage of this research as well as for giving me extraordinary experiences throughout the work. Above all, and the most needed, he provided me unflinching encouragement and support in various ways. His scientist's intuition has made him a constant oasis of ideas and passions in science, which exceptionally inspire and enrich my growth as a student, a researcher, and as a scientist-want-to-be. I could never have embarked upon and started all of this without his prior teachings in Digital Signal Processing and its applications which opened up unknown areas to me. I am indebted to him more than he knows. Thank you.

I gratefully acknowledge Dr. Michael Harrison for his advice, supervision, and crucial contribution, which made him a backbone of this research and to this thesis. His involvement with his originality has triggered and nourished my intellectual maturity that I will benefit from for a long time to come. I am grateful in every possible way and hope to continue our collaboration in the future.

My sincere thanks also go to Professor Ahmed Al-Jumaily, the role model for hard workers in the IBTec, AUT. I would like to thank him for being the first person who taught me how to work in a professional environment. I am proud to record that I had several opportunities to work with such an exceptionally experienced scientist.

Many thanks go, in particular, to Dr. Andrew Lowe for timely support, guidance, and for providing technical expertise with his DOMonitor application which smoothed the entire data conversion and testing phase of the project, and to Bhupendra Gohil for his valuable advice and furthermore, his precious weekend times that were used in some critical issues of this research, and to the guidance and excellent support from my colleagues at IBTec, AUT. I have also benefited from advice and guidance from them.

Where would I be without my family? My parents deserve special mention for their inseparable support and prayers. My father, Ibrahim Mirza, in the first place is the person who built the foundation for my learning character as he showed me the joy of intellectual pursuit since my childhood. My mother, Dr. Iffath Unnisa, is the one who

sincerely raised me with her caring and gentle love. To my brother, Mohsin Mirza and his wife, Sadaf Ansari, thank you for continuously supporting me throughout my life and leaving me free in all my decisions. I would also like to thank Mohsin for his technical support whenever I needed it.

Finally, I would like to thank everybody who was important to the successful realization of this thesis, as well as expressing my apology for not mentioning each one personally.

Abstract

Man has a limited ability to accurately and continuously analyse large amounts of data. Observers are typically required to monitor displays over extended periods and to execute overt detection responses to the appearance of low probability critical signals. The signals are usually clearly perceivable when observers are alerted to them, but they can be missed in the operating environment.

The challenge is to develop a computer application that will accumulate information on a variable, or several variables, over time and identify when the trend in observations has changed. In recent years, there has been a rapid growth in patient monitoring and medical data analysis using decision support systems, smart alarm monitoring systems, expert systems and many other computer aided protocols.

The expert systems have the potential to improve clinician performance by accurately executing repetitive tasks, to which humans are ill-suited. Anaesthetists working in the operating theatre are responsible for carrying out a multitude of tasks which requires constant vigilance and thus a need for a smart decision support system has arisen. The decision support tools capable of detecting pathological events can enhance the anaesthetist's performance by providing alternative diagnostic information.

The main goal of this research was to develop a clinically useful diagnostic alarm system using two different techniques for monitoring a pathological event during anaesthesia. Several techniques including fuzzy logic, artificial neural networks, control and monitoring techniques were explored.

Firstly, an industrial monitoring system called Supervisory Control and Data Acquisition (SCADA) software is used and implemented in the form of a prototype system called SCADA monitoring system (SMS). The output of the system in detecting hypovolaemia was classified into three levels; mild, moderate and severe using SCADA's InTouch software. In addition, a new GUI display was developed for direct interaction with the anaesthetists.

Secondly, a fuzzy logic monitoring system (FLMS) was developed using the fuzzy logic technique. New diagnostic rules and membership functions (MF) were developed using MATLAB. In addition, fuzzy inference system FIS, adaptive neuro fuzzy inference system ANFIS and clustering techniques were explored for developing the FLMS's diagnostic modules.

The raw physiological patient data acquired from an S/5 monitor were converted to a readable format using the DOMonitor application. The data was filtered, preprocessed, and analysed for detecting anaesthesia related events like hypovolaemia. The accuracy of diagnoses generated by SMS and FLMS was validated by comparing their diagnostic information with the one provided by the anaesthetist for each patient. Kappa-analysis was used for measuring the level of agreement between the anaesthetist's, SMS's, and FLMS's diagnoses.

In offline analysis both systems were tested with data from 15 patients. The SMS and FLMS achieved an overall agreement level of 87 and 88 percent respectively. It implies substantial level of agreement between SMS or FLMS and the anaesthetists.

These diagnostic alarm systems (SMS and FLMS) have shown that evidence-based expert diagnostic systems can diagnose hypovolaemia, with a substantial degree of accuracy, in anaesthetized patients and could be useful in providing decision support to anaesthetists.

Table of Contents

List of Figures

List of Tables

Statement of Originality

‘I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning. It contains results of my investigation, except where otherwise stated. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.’

I understand that my thesis may be made electronically available to the public.

Signed: ……

Date: …….

List of Abbreviations

AHV	Absolute Hypovolaemia
ANFIS	Adaptive Neuro Fuzzy Inference System
ANN	Artificial Neural Networks
BP/P1	Invasive Blood Pressure
CTS/RTS	Clear to Send /Request to Send
DTFT	Discrete-Time Fourier Transform
$ETCO_2$	End Tidal (exhaled) Carbon-Dioxide
FCM	Fuzzy C-Means clustering
FCO	Fall in Cardiac Output
FFT	Fast Fourier Transform
FIS	Fuzzy Inference System
FLMS	Fuzzy Logic Monitoring System
GUI	Graphical User Interface
HR	Heart Rate
I/O	Input/Output
ICO_2	Inspired (inhaled) Carbon-Dioxide
IDAS	Injectable Drug Administration and automation System
K	Kappa value
LAN	Local Area Network
MAP	Mean Arterial Pressure
MF	Membership Function
MH	Malignant Hyperpyrexia
mmHg	Millimetres of mercury
MV	Minute Volume
NEG	Negative
OLE	Object Linking and Embedding
OT	Operating Theatre
P2	Central Venous Pressure

PAW	Peak Airway Pressure
PLC	Programmable Logic Controllers
POS	Positive
PV	Pulse Volume / Plethysmography
RHV	Relative Hypovolaemia
RR	Respiratory Rate
RT	Real Time Alarm
RT-SAAM	Real-Time Smart Alarms for Anaesthesia Monitor
RTU	Remote Telemetry Units
S/5	Datex-Ohmeda, GE S/5 anaesthesia monitor
SAAM	Smart Alarms for Anaesthesia Monitoring
SAP	Systolic Arterial Pressure
SCADA	Supervisory Control and Data Acquisition
SD	Standard Deviation
SMS	SCADA Monitoring System
SPV	Respiration-induced systolic pressure variations
SV	Stroke Volume
TV	Tidal Volume
WAN	Wide Area Network

CHAPTER 1 Introduction

1.1 Background

1.1.1 Biomedical Research

Biomedical research has emerged as a basic requirement for the development of health care and medicine. Without research we would face a great deal of difficulty mitigating even a common cold!

The purpose of biomedical research is to provide solutions to health risks and disease with minimal or no additional risks.

To acquire the knowledge necessary to understand life processes, medical researchers conduct biological experiments to formulate hypotheses and predictions. They test the accuracy of their predictions by conducting further experiments and make changes in their hypotheses based on the results of these experiments. This process of predicting and experimenting continues – and, often, new insights evolve that need further probing.

Each ‘breakthrough’ is actually the result of countless hours, days, and years of work. Every investigation involves several bits of information derived from many different research specialities. When these bits of information are combined, they often provide a solution to a health problem. Even if the problem is not completely solved, the information gleaned provides a deeper understanding of the scientific process so the search for a solution can continue.

1.1.2 Anaesthesia

The word 'anaesthesia' means 'loss of sensation'. Today, safe and effective methods of anaesthesia allow pain-free surgery to be performed on millions of patients every year.

There are a few important things about anaesthesia:

- Above all, it prevents the feeling of pain and other sensations during an operation.

- It can be administered in various ways.

- Not all anaesthesia make the patient unconscious.

- It can be targeted to specific parts of the body.

Drugs that cause anaesthesia work by blocking the signals that pass along nerves to the brain. When the drugs wear off, the person starts to experience normal sensations again, including pain. Anaesthesia may be applied to the whole body; this is known as general anaesthesia, or to a specific part of the body, which is known as regional or local anaesthesia. All of these techniques involve giving particular drugs that interfere with the transmission of nervous impulses in order to reduce sensation. 'Anaesthetic' is the term applied to some or all of the drugs used to produce 'anaesthesia', and is also used to describe the whole process.

1.1.2.1 General Anaesthesia

General anaesthesia is defined by three main components: unconsciousness, analgesia, and immobility; the drugs used to maintain these components are the hypnotics, analgesics, and muscle relaxants, respectively. This type of anaesthesia makes the person unconscious and insensitive to pain because the agents used affect either the brain or the nerves. These medicines are given intravenously or through a mask. During this artificial sleep, the anaesthetist observes, monitors, and supports the vital functions of the body.

1.1.2.2 Local Anaesthesia

This anaesthetic technique blocks the nerves that transmit the pain; as a result, the area served by these nerves becomes desensitised. Local anaesthetics are used for this purpose. The name of this technique varies according to the part of the body that is to be anaesthetised.

1.1.3 Anaesthesiologist

In an Operating Theatre (OT), the anaesthesiologist is required to manage various responsibilities simultaneously: the patient's physiological monitors and audio alarms, their fluid needs and drug administration, as well as student education. Although the anaesthesiologists carefully prioritise these tasks, the cumulative need may still exceed the limit of even highly trained, focused clinicians [1]. Technology, in the form of an expert system, can convey more precise information about a patient and potentially improve vigilance, standardise clinical protocols, enhance situational awareness, and reduce errors in anaesthetic practice [2]. Computers have the capacity to monitor large volumes of diverse data rapidly, while humans are only able to monitor a maximum of seven different parameters at any given time [3].

Hospital intensive care areas generate enormous amounts of real-time and off-line data related to the status of acutely ill patients: multi-parameter real-time physiological signals, ventilator data, laboratory tests, imaging studies, medications, clinical observations, etc. [4]. Anaesthesiologists must make a variety of decisions, in real-time, based on inputs from countless monitoring instruments in much the same way as pilots of high performance aircraft or operators of nuclear power plants must do [5]. Human errors contribute to a large portion of the anaesthesia-related mishaps; these could easily be prevented by providing decision support to the anaesthetists [6, 7].

Hundreds of computerised decision support systems (DSS) or expert systems and other aids have been developed to assist patient management in order to enhance the anaesthetists' performance [8, 9].

1.2 Medical Literature

The human body is a complex living machine, and to cover the minute details of its structure and function is beyond the aim of this text; in this section we focus our study on the important physiological parameters related to our research, namely, the cardiovascular system.

1.2.1 The Cardiovascular System

The heart and the circulatory system are the main components of the cardiovascular system. The cardiovascular system is comprised of the heart, blood vessels or vasculature, and the cells and plasma that make up the blood. The blood vessels of the body represent a closed delivery system which transports blood around the body, circulating substances such as oxygen, carbon dioxide, nutrients, hormones, and waste products. There are three principal types of blood vessels:

- Veins - the afferent blood vessels that return blood to the heart.
- Arteries - the efferent blood vessels that carry blood away from the heart.
- Capillaries - narrow, thin-walled blood vessels that form networks within the tissues.

Arteries branch and diverge as they move away from the heart. They form smaller and smaller divisions until they eventually terminate in capillaries. By contrast, veins merge and converge into successively larger blood vessels as they move towards the heart. Capillary networks are the sites of gas, nutrient, and waste exchange between the blood and the respiring tissues.

1.2.1.1 The Human Heart

The heart weighs between 7 and 15 ounces (200 to 425 grams) and is a little larger than the size of one’s fist. By the end of a long life, a person's heart may have beaten (expanded and contracted) more than 3.5 billion times. In fact, each day, the average heart beats 100,000 times and pumps about 2,000 gallons (7,571 litres) of blood.

The heart is located between the lungs in the middle of the chest, behind and slightly to the left of the breastbone (sternum). A double-layered membrane called the pericardium surrounds the heart like a sac. The outer layer of the pericardium surrounds the roots of the heart's major blood vessels and is attached by ligaments to the spinal column, diaphragm, and other parts of the body. The inner layer of the pericardium is attached to the heart muscle. A coating of fluid separates the two layers of membrane, letting the heart move as it beats, yet still be attached to the body.

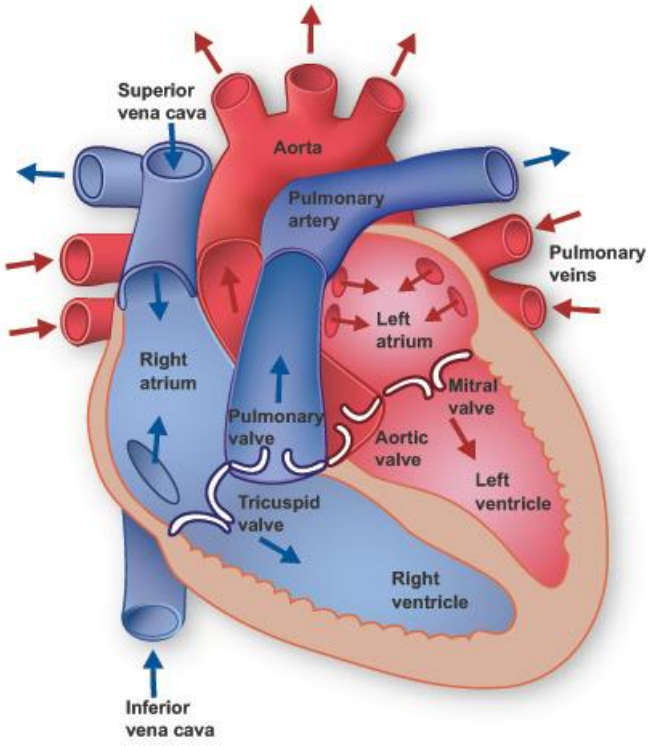


Figure 1.2.1-1. Human heart.

The human heart has four chambers. The upper chambers are called the left atrium and the right atrium; the lower chambers are called the left ventricle and the right ventricle. A wall of muscle, the septum, separates the left and right atria and the left and right ventricles. The left ventricle is the largest and strongest chamber in the heart, its walls are only about a half-inch thick, but they have enough force to push blood through the aortic valve and into the body [10].

1.2.1.2 The Heart Valves

Four types of valves regulate blood flow through the heart:

- The tricuspid valve regulates blood flow between the right atrium and right ventricle.
- The pulmonary valve controls blood flow from the right ventricle into the pulmonary arteries, which carry blood to the lungs to pick up oxygen.
- The mitral valve lets oxygen-rich blood from the lungs pass from the left atrium into the left ventricle.

- The aortic valve opens the way for oxygen-rich blood to pass from the left ventricle into the aorta, the body's largest artery, from where it is delivered to the rest of the body.

1.2.1.3 The Circulatory System

The one-way circulatory system carries blood to all parts of the body. This process of blood flow within the body is called circulation. Arteries carry oxygen-rich blood away from the heart, and veins carry oxygen-poor blood back to the heart. In the pulmonary circulation, though, the roles are switched. It is the pulmonary artery that brings oxygen-poor blood into the lungs and the pulmonary vein that brings oxygen-rich blood back to the heart [11].

1.2.2 Regulation of Blood Pressure

Constant and adequate pressure in the arterial system is required to drive blood into all of the organs. Abnormally low blood pressure results in inadequate perfusion of organs, while abnormally high blood pressure can cause heart disease, vascular disease, and stroke. It is essential that blood pressure be maintained within a narrow range of values that is consistent with the needs of the tissues.

Pressure in the arterial system fluctuates with the cardiac cycle. Blood pressure reaches a peak in systole and is lowest in diastole. Rather than focusing on these extremes of blood pressure we will discuss blood pressure in terms of the Mean Arterial Pressure (MAP). Mean arterial pressure represents the average pressure in the arterial system. This value is important because it is the difference between MAP and the venous pressure that drives blood through the capillaries of the organs. Because more time is spent in diastole than in systole, MAP is not simply the average of the systolic and diastolic pressures.

A simple formula for calculation of MAP is:

MAP = diastolic pressure + 1/3 pulse pressure

Pulse pressure = systolic pressure - diastolic pressure

1.2.2.1 Major factors that affect Mean Arterial Pressure

The three most important variables affecting MAP are:

- Total peripheral resistance (TPR)
- Cardiac output
- Blood volume

MAP = Cardiac Output X Total Peripheral Resistance

1.2.2.2 Total Peripheral Resistance (TPR)

Blood vessels provide resistance to the flow of blood because of friction between moving blood and the wall of the vessel. The TPR refers to the sum total of vascular resistance to the flow of blood in the systemic circulation. Because of their small radii, arterioles provide the greatest resistance to blood flow in the arterial system. An adjustment in the radii of arterioles has a significant effect on TPR, which in turn has a significant effect on MAP. Resistance and pressure are directly proportional to each other. If resistance increases, then pressure increases. When the radii of arterioles decrease with vasoconstriction, TPR increases, which causes MAP to increase.

1.2.2.3 Cardiac Output

Cardiac output is the volume of blood pumped by the heart per minute (ml blood/min); it is a function of heart rate and stroke volume. The heart rate is simply the number of heart beats per minute. The stroke volume is the volume of blood, in millilitres (ml) pumped out of the heart with each beat. Increase in either heart rate or stroke volume increases cardiac output.

Cardiac Output in ml/min = heart rate (beats/min) X stroke volume (ml/beat).

An average person has a resting heart rate of 70 beats/minute and a resting stroke volume of 70 ml/beat. The cardiac output for this person at rest is:

Cardiac Output = 70 (beats/min) X 70 (ml/beat) = 4900 ml /minute.

The total volume of blood in the circulatory system of an average person is about five litres (5000 ml). According to our calculations, the entire volume of blood within the circulatory system is pumped by the heart each minute (at rest). During vigorous exercise, the cardiac output can increase up to seven-fold (35 litres/minute).

Hence, blood flow is directly proportional to pressure (Flow = pressure/resistance), therefore an increase in flow (cardiac output) will cause an increase in pressure (MAP) [12].

1.2.2.4 Blood Volume

Blood volume is directly (but not linearly) related to blood pressure. If the blood volume is increased, then venous return of blood to the heart will increase. An increase in venous return will, by Starling's Law, cause stroke volume to increase. As stroke volume goes up, the cardiac output goes up and the blood pressure rises. Thus, one way to control blood pressure over the long-term is to control blood volume. There may, however, be interacting systems; perfusion is more important than absolute blood pressure.

1.2.2.5 End Tidal Carbon-Dioxide (ETCO2)

Carbon-dioxide (CO_2) produced by cellular metabolism is transported to the right ventricle by the venous system. Hence, it is pumped to the lungs by the heart and diffuses into the exhaled air. The concentration at the end of the expiration cycle is called End Tidal CO_2 ($ETCO_2$), which reflects the metabolism, circulation and ventilation [13].

1.2.2.6 Inspired carbon-dioxide (ICO2)

Inspired (fractional) carbon-dioxide ($FICO_2$) is the carbon-dioxide breathed in by the patient and is normally zero. If re-breathing occurs then the $FICO_2$ may be greater than zero. A raised CO_2 will affect the patient's hemodynamic state [14].

These parameters can be used for monitoring the patient's hemodynamic state during anaesthesia; detailed work using some of these parameters is discussed in the following chapters of this thesis.

1.3 Motivation

The main objective of this research is to develop a diagnostic alarm prototype capable of generating diagnosis for monitoring anaesthesia. The work towards this objective has been carried out with the following approaches:

1.3.1 Supervisory Control and Data Acquisition (SCADA)

Use of the industrial software called Supervisory Control and Data Acquisition (SCADA) to diagnose hypovolaemia in a patient under anaesthesia; it is called the SCADA Monitoring System (SMS).

1.3.2 Fuzzy logic Monitoring System (FLMS)

Use of Fuzzy Logic to diagnose hypovolaemia in a patient under anaesthesia; this is called Fuzzy Logic Monitoring System (FLMS).

1.4 Literature Review

Over the past two decades, computers have played an important part in research on the clinical decision-making process. Over the next two decades, computers will undoubtedly provide major information and advancement in the practice of medicine. Szolovits et al. [15] in the 1960's, and Shortliffe et al. suggested a coherent summary of computer-aided decision making in medicine at that time [16].

Humans are limited in their ability to accurately and continuously analyse large quantities of data simultaneously. The challenge is to develop a computer application that will accumulate all the information on a variable, or several variables, over time, and identify when the trend in observations has changed. In recent years, there has been rapid growth in this patient monitoring approach by Decision Support Systems (DSS), Smart Alarm monitoring systems, expert systems, and many other computer-aided protocols and designing tools [17-22]. In this project we focus on expert systems and decision support systems applied to anaesthesia, and their applications for monitoring pathological conditions during the administration of anaesthesia.

1.4.1 Need of Expert Systems in Anaesthesia Monitoring

Health care professionals often rely on their knowledge and experience together with observations and measurements to make clinical judgments and plan appropriate therapy regimes. In many circumstances, however, intuitive decision-making fails to achieve the optimal balance under competing or conflicting care delivery demands. In certain circumstances an anaesthetist may be overloaded by patient data from the monitoring equipment and the analysis of this data. The anaesthetist could possibly be

too involved with this data to continuously diagnose the patient's condition [23] and this could lead to potentially grave consequences.

Human errors in anaesthesia account for more than 80 percent of the preventable mishaps [7]. Computers have the capability to monitor large volumes of diverse data rapidly, whereas humans are only able to monitor a maximum of seven different parameters at any one time[3]. Van den Eijkel et al. suggested that such limitations could be overcome by using a knowledge-based anaesthesia monitor (Expert System) which is an evidence/knowledge-based system that analyses the data and presents the results to the anaesthetist; this could potentially prevent such mishaps [6].

1.4.2 Decision Support Systems in Anaesthesia Monitoring (DSS)

Decision Support Systems (DSS) are an emerging and potentially beneficial technology that shows great promise for reducing medical errors and delivering improvements in the quality, safety, and efficiency of health care. Decision Support Systems are designed to integrate a patient monitoring information system: a computerised medical knowledge base and an inference engine to generate case-specific and situation-specific advice [24, 25]. These systems should also help the anaesthetists make decisions in complex situations where continuous monitoring of highly critical physiological parameters such as BP, HR, ETCO2, and SAP require immediate response [26]. Following are the brief reviews in this area of monitoring using the Decision Support Systems:

Dunsmuir et al. [27] developed software for use as a knowledge authoring tool for physiological monitoring in anaesthesia. This application enables clinicians to create knowledge rules without the need for a skilled computer engineer or programmer. The rules are designed to provide clinical diagnoses, explanations, and treatment advice to the clinician in real time for optimal patient care. The knowledge base consists of a set of rules, and each rule is designed as an IF-THEN statement that represents expert knowledge. Each rule consists of a list of patterns and an outcome. Each pattern is a statement about a data parameter; each will be true or false, depending on the value of the data parameter. The outcome is a descriptor of the patient's status that exists given that all the rule's patterns are true. By intelligently combining data from physiological monitors and demographic data sources, the expert system can use these rules to assist

in monitoring the patient. The Post-Study System Usability Questionnaire (PSSUQ) was used to record feedback on usability of the DSS; the result showed strong agreement of about 80 percent.

Mahfouf et al. [28] propose a fuzzy logic-based Decision Support System (DSS) for the management of post-surgical cardiac intensive care unit (CICU) patients. The decision support system includes three building blocks: a **Hemodynamic Classifier** as an input module to evaluate the patient's hemodynamic status, a **Therapy Adviser –** a diagnostic module which implements the expert decision-making strategies, and a **Multiple-Drug Fuzzy Controller** – a therapeutic module which incorporates a multiple-drug fuzzy control system for the execution of the therapeutic recommendations which interact with the patient and clinician. As a result, this system shows acceptable agreement towards the therapeutic recommendations.

1.4.3 Expert Systems or Smart Monitoring Systems in Anaesthesia Monitoring

Technology designed to assist anaesthesiologists in improving and maintaining vigilance during monitoring tasks represents a potential mechanism by which anaesthetic care can be improved. Several studies concluded that computerized monitoring has significant potential to improve clinical monitoring performance[29]. Observers are typically required to monitor displays over extended periods, and to execute overt detection responses to the appearance of low probability critical signals. The signals are usually clearly perceivable when observers are alerted to them, but they can be missed in the operating environment [30]. Hence, the expert systems have the potential to improve clinician performance by accurately executing repetitive tasks to which humans are ill-suited, such as physiological parameter analysis and surveillance. Additionally, expert systems can be used to standardize clinical guidelines, and provide memory aids for the clinician [31]. Expert systems have been developed to aid clinicians with their decision-making with respect to patient care in various fields of medicine. Such systems are discussed below.

Bhupendra et al. [32] developed the diagnostic system, Real Time – Smart Alarms for Anaesthesia Monitoring (RT-SAAM); it has two modules: probabilistic alarms (Probabilistic Module) and respiration-induced systolic pressure variations (SPV Module). These were developed using Matlab™ and Labview™. The accuracy of the

diagnostic results from RT-SAAM was analyzed using Kappa analysis. The results showed that the developed diagnostic system (RTSAAM) is capable of diagnosing the pathological events with a substantial-to-fair level of agreement between RTSAAM and the anaesthetist.

Ansermino et al. [33] developed a software tool (iAssist) to support clinicians as they monitor the physiological data that guides their actions during anaesthesia. The system tracks the statistical properties of multiple dynamic physiological processes, and identifies new trend patterns. They report their initial evaluation of this tool (in pseudo real-time) and comparing the detection of trend changes to a post hoc visual review of the full trend. The Cumulative Sum (CUSUM), technique demonstrates the differences when successive observed values and another target value (a predicted value in this case) are accumulated. This technique is used in this study to detect changes in Heart Rate (HR), End Tidal Carbon Dioxide (ETCO2), Exhaled Minute Ventilation (MVexp) and Respiratory Rate (RR); it is also used to predict future observations and update the CUSUM threshold in real-time. As a result, the software tool (iAssist) correctly identified 869/960 changes at a rate of 12.9 per hour.

Gunther et al. [34] developed a respiratory variation (RV) monitoring device (RV monitor). It continuously records both airway pressure and arterial blood pressure (ABP) and then compares the RV monitor measurements with manual Difference in Pulse Pressure (DPP) measurements using two methods: a Strip Chart method and the Point and Click method. The result of this study concluded that the Point and Click Method is preferable.

1.4.4 Fuzzy Systems – fuzzy logic based systems in Anaesthesia Monitoring

Computer programs employing *fuzzy logic* are intended to imitate human thought processes in complex circumstances, but to function at greater speed [35]. Fuzzy logic-based expert systems have been developed in each and every area of anaesthesia monitoring. The work towards the **Control of anaesthetic gases and blood pressure** by Sieber and colleagues [36] reported accurate control of mean alveolar concentration of isoflurane by a system that altered the gas flow rates. **Postoperative pain control** resulting in the patient's target analgesia level achieved a positive

conclusion as much as 77 percent of the time, as reported by Carregal et al. [37], and there are many more such examples.

Lowe [38] has developed a system called SENTINEL; this is computer software that could help in fault detection and diagnosis (FDD) in anaesthesia. It detects problems that can occur during anaesthesia by analysing physiological signals on-line. The advantage of this system over others is that it provides a measure of objectivity and vigilance. It uses a fuzzy time-domain pattern matching technique, termed *fuzzy trend templates*, to detect vaguely specified patterns in multiple physiological data streams. These patterns are representative of symptoms associated with undesirable patient states. As an expert system, it incorporates the knowledge of many consultant anaesthetists. Sentinel has achieved sensitivity and specificity accuracy above 90 percent in the diagnosis of seven common or serious conditions that can arise during anaesthesia.

Lowe and Harrison [39] developed a fuzzy logic based algorithm for detecting a rare pathological condition called malignant hyperpyrexia (MH). In this study, rule-based diagnoses are performed to detect the changes in the patterns of symptoms. In offline validation of the algorithm, the system detected MH nine minutes before the anaesthetist diagnosed it. The work and its proven results show how expert systems can be implemented to facilitate and enhance anaesthetists' performance in the clinical environment, thus improving patient safety.

Esmaeilia et al. [40] designed a fuzzy rule based system which integrates the features of an electroencephalogram (EEG) to quantitatively estimate the depth of anaesthesia (DOA). Reference data is divided into four well-defined anaesthetic states: awake, moderate anaesthesia, surgical anaesthesia, and isoelectric (deeply unconscious). Statistical analysis of features was used to design input Membership Functions (MFs). The training data was used in Adaptive Network-based Fuzzy Inference System (ANFIS) to label the partitions and extract efficient fuzzy IF-THEN rules; the Fuzzy Rule-Base Index (FRI) is calibrated between 0 (isoelectric) and 100 (fully awake) using the Fuzzy Inference System Engine (FIS) and designed output MFs. The main focus in this study is to simplify the mutual knowledge exchange between the human

expert and the machine, and lead to enhancement of both the interpretability of the results and the performance of the system.

Mahfouf et al. [41] developed a model into a Mamdani type of fuzzy model using anaesthetists' knowledge described by fuzzy IF-THEN rules. Clinical data are used to construct the patient model. The effect of the actual concentrations of anaesthetic and analgesic drugs was used to model the pharmacodynamic interactions of the two drugs on the cardiovascular parameters, and on the auditory evoked potentials. An Adaptive Network-based Fuzzy Inference System (ANFIS) was then used to train fuzzy Takagi–Sugeno–Kang (TSK) models so as to describe the different signals. A stimulus model was used to establish the effects of the surgical stimulus on Heart Rate (HR) and systolic arterial pressure (SAP) according to the level of analgesia used to model the different signals.

1.4.5 Statistics based systems in Anaesthesia Monitoring

The changes in physiological variables during anaesthesia are a combination of responses to drugs and surgery [42]. This diagnostic technique is used for the detection of complex changes in physiological variables, and responds in a more clinically appropriate manner than the normal threshold alarm systems.

Harrison and Connor [43] developed an anaesthesia alarm system that detects the changes in SAP and states that a decrease in systolic arterial pressure of 10 mmHg from a previous value of 70 mmHg has a greater clinical significance than a decrease of 10 mmHg from 150 mmHg. They have processed systolic arterial pressure data to create a mathematically straightforward statistical tool for sampling intervals up to five minutes.

Using Pythagoras's theorem, they combined the value for the standard deviation of systolic arterial pressure and the standard deviation of the change in systolic arterial pressure, so instead of alarms being set in mmHg, they would be set in standard deviations. This technique was developed further using Principal Component Analysis to isolate uncommon deviations from normal, clinically unimportant, physiological variations. This may turn out to be clinically useful.

Dosani et al. [44] has developed a system which detects the change in the physiological trend and warns the clinician. Using the physiological parameters like end-tidal carbon dioxide, heart rate, exhaled minute ventilation, non-invasive arterial pressure, respiratory rate, and oxygen saturation, the algorithms were tested. A Kalman filter is used in the algorithms to predict future observations and update the Cumulative Sum (CUSUM) threshold in real-time during anaesthesia. This provides context sensitive monitoring and includes real-time feedback on each change point with the completion of a Post-Study System Usability Questionnaire (PSSUQ). As a result of this experiment, the algorithms detected 22 change points per surgical case, more than 60 percent of which were considered clinically useful and fewer than 7 percent of which were due to artefacts.

1.4.5.1 Target Control Systems in Anaesthesia Monitoring

The target control systems originate from pharmacokinetic models. Drug distribution and effect are described using pharmacokinetics and pharmacodynamics. The pharmacokinetics are often described as "what the body does to the drug" and pharmacodynamics as "what the drug does to the body" [45, 46]. Recently, using those models, an anaesthesia monitoring system has been developed; a review of the work follows.

Bressan et al. [47] developed software for data acquisition and control (ASYS) in a clinical setup. Similar to the industrial Supervisory Control and Data Acquisition (SCADA), the software assembles Target Controlled Infusion (TCI) monitoring and supervisory control data in real time from devices in a surgical room. Based on pharmacokinetic models, the effect-site and plasma concentrations can be related to the drug dose infused and vice versa. The software determines the infusion rates of the drug, and sends commands to the infusion pumps. This software provides the anaesthesiologist with a trustworthy tool for managing a safe and balanced anaesthesia since it also incorporates the acquisition and display of the patient's brain signals. The clinical results showed that ASYS is a promising tool in the behaviour of anaesthesia, when compared with the standard method. The synchronization anaesthesia software performed well in its first clinical experiments in animal anaesthesia.

1.4.5.2 Proportional Derivative Control System in Anaesthesia Monitoring

The anaesthesia system, the patient, and the anaesthesiologist, have been compared to a closed-loop control system. Applying this concept, the limit alarm is similar to a component of a servo control algorithm, as it gives feedback to the anaesthesiologist [48]. Closed loop control systems usually use a more sophisticated processing of input signals to enable appropriate grading of the response, and to prevent overreaction to irrelevant changes in the target variable under control. More sophisticated ways of implementing closed loop control systems include Proportional-Integral-Derivative (PID) control, adaptive control, and fuzzy logic control; one of these systems is described below.

James et al. [49] developed an alarm system based on a modified proportional-derivative (PD) controller algorithm, and prospectively tested its ability to predict significant hypotensive episodes. The alarm algorithm was tuned to detect hypotension using selected invasive arterial pressure and the performance was tested prospectively in comparison to conventional limit alarms and median filtered limit alarms, set at 85 mmHg and 90 mmHg, to assess its ability to predict hypotensive episodes. As a result of this PD alarm algorithm, it was reported that the onset and offset times for significant hypotensive episodes were similar. The false positive rate was 34 percent compared with 45–64 percent for the other alarms ($p < 0.01$). Hence this arterial pressure alarm system design, based on a closed loop control algorithm, showed improved performance over conventional limit alarms.

The development of systems to enhance anaesthesia monitoring is growing in all areas, as discussed above; monitoring is now performed continuously, using highly sophisticated machines and techniques [50]. These systems give the output as alerts or alarms, in terms of sound, light, flash, and display; however, it has quickly become evident that the surplus of alarms competing for the attention of the anaesthetist has a negative impact on their performance [51] and the high rate of false alarms has resulted in anaesthetists working with most alarms disabled or in silent mode in order to avoid distraction[52]. There have been some early attempts made to reduce false alarms in anaesthesia monitoring [53-55]. Some such systems and works in development are discussed in following section.

1.4.6 Alarms in Anaesthesia Monitoring

The alarms of medical devices are a matter of concern in critical and preoperative care. The frequent false alarms are not only a nuisance for patients and caregivers, but can also compromise patient safety and the effectiveness of care. Intensive care unit (ICU) alarms have been designed to call attention to the patient, to alert a change in their physiology, or to alert staff to a device problem. Alarms are triggered when a physiologic variable crosses a set threshold. In their excellent literature review, Imhoff and Kuhls report alarm frequencies of 1.6 to 14.6 alarms/h and a false alarm rate of up to 90 percent [56]. Chambrin et al. [57] reported the lowest rate of alarms at 1.6 alarms/h. Tsien and Fackler [58] reported one of the highest alarm rates at 9.8 alarms/h in the noisier environment, but limited their study to alarms from the cardiac patient monitor. The problem with simple threshold alarms is that up to 94.5 percent of the alarms that sound in the ICU are either false or provider-induced, [59] and frequently sound unnecessarily [56, 57, 59]. Default settings by the equipment manufacturers are set to avoid missing a single false negative alarm, and thereby result in many false positive alarms [60].

1.4.6.1 Reasons and Classification of False Alarms

Technical problems or artefacts can cause false alarms. Another reason for false alarms is the clinical inappropriateness of the alarm itself, the alarm limit, or the alarm algorithm.

From a medical perspective, there are three categories of false alarms:

- **Technically false alarms** are situations in which an alarm is called for a specific variable, although this variable, in reality, does not surpass any preset threshold. Examples are an asystole call because of a low voltage lead, motion artefacts, falsely low SpO_2 readings in hypothermic patients, or a wet flow sensor of a ventilator.

- **Clinically false alarms** are situations in which the variable is actually beyond the preset alarm threshold, but the situation does not always have clinical relevance. Examples are: the intermittently increased heart rate in patients with an arrhythmia, when the increased heart rate may only last for a few

seconds, or during weaning from mechanical ventilation when a patient takes a deep spontaneous breath exceeding the tidal volume alarm limit.

- **False alarms** through interventions are actually both technically and clinically false alarms. They are caused by medical or nursing staff interventions such as moving or positioning the patient, drawing blood, and flushing the a-line, or disconnecting the patient from the ventilator for endotracheal suctioning, as suggested and discussed in the excellent review by Imhoff and Kuhls [56].

1.4.7 Methods designed to reduce False Alarms in Anaesthesia Monitoring

There is continuing interest in the development of intelligent alarms and expert systems that addresses the serious problem of false alarms [61]; such expert systems combined with best knowledge and robust technique give excellent performance [29]. In recent work by Matthias et al. [62] there was a decrease of 67 percent of the false alarms just by introducing a 19-second delay. The physiological parameters taken in this study were TV, MV, HR, BP, RR, PAW, and InfP. The alarms were classified as Effective patient, Effective technical, ignored, and ineffective. In the 2344 tasks which were performed in this study, 1214 alarms occurred. On average, alarms occurred 6.07 times per hour and were active for 3.28 minutes per hour; 23 percent were effective, 36 percent were ineffective, and 41 percent were ignored; the median alarm duration was 17 seconds. As a result, the researchers suggested that a 14-second delay before alarm presentation would remove 50 percent of the ignored and ineffective alarms, and a 19-second delay would remove 67 percent and reduce the number of ineffective and ignored alarms from 934 to 274.

Rheineck-Leyssius and Kalkman [63] proposed a highly effective method for reducing pulse oximeter (SpO_2) alarms by introducing a 6-second delay, thereby reducing alarm rates by 50 percent. The authors compared the effects of different methods on the number of true and false alarms: alarm delay from 2 s to 44 s with an alarm limit set to 90 percent, a mean and median filter by 10 s to 90 s and decreasing the alarm limit by 90 percent.

Zong et al. [64] developed an algorithm for reducing false alarms related to changes in arterial blood pressure (ABP). The algorithm assesses the ABP signal quality, analyses the relationship between the electrocardiogram and ABP using a fuzzy logic

approach, and post-processes (accepts or rejects) ABP alarms produced by a commercial monitor. The algorithm was developed and evaluated using unrelated sets of data from the MIMIC database. As a result, the system rejected 98.2 percent (159 of 162) of the false ABP alarms and reduced the false ABP alarm rate from 26.8percent to 0.5percent of ABP alarms, while accepting 99.8 percent (441 of 442) of true ABP alarms.

The use of median filtering techniques seems to be effective in decreasing the number of false alarms for data coming from the ventilator [65] as well as those coming from the cardiovascular monitor [66]. Schoenberg et al. [67] developed a trend detection algorithm for the ICU setup using the parameters HR, SAP, and SpO2. The alarms are classified by the clinicians, the standard system, as true and false; the new trend detection algorithm-based system is applied to the same data, and as a result the new algorithm achieved 82 percent of sensitivity.

Most of the studies seek to reduce the number of false alarms by using multi-parameter approaches, and the knowledge of the experts in the field is then used to determine the episodes of artefacts or specific events. Many such studies have been conducted in this way [67-71].

Based on the literature presented above, we wanted to develop a system that has more clinical relevance, has the ability to help the clinician, and can do this in a more user-friendly manner. This work involved using two different techniques: fuzzy logic with the capability to handle the complex build and good results (as discussed above), and the widely used industrial software called SCADA.

1.4.8 Summary of Literature Review

Below is the table which summarises the work discussed above, gives the details of the work done by the researchers with different methods and techniques, and summaries their results.

Table 1.4.8-1. Summary of Literature Review.

Study	Technique/method used	Parameters used	Investigation topic	Study results
Dunsmuir et al. [27]	Knowledge authoring tool using Fuzzy logic based rules with the PSSUQ	HR, BP and age	Malignant hyperthermia	PSSUQ shows the strong agreement of about 80% positivity
Mahfouf et al. [28]	Fuzzy logic based Decision Support System (DSS) for multi drug fuzzy controller	SBP, CVP, SVR, CO and HR.	Hypertension, hypotension, hypovolaemia and septic shock	Response of multi drug advisor were accurate and acceptable
Bhupendra et al. [32].	RT-SAAM using Fuzzy Module, Probabilistic Module and SPV Module	HR, SAP, MAP, ETCO2, RR, BP, PV.	Different physiological events	Substantial to fair level of agreement between RTSAAM and the anaesthetist
Ansermino et al. [33]	iassist software tool using CUSUM technique	HR, ETCO2, MVexp, RR	Monitor the physiological data	iassist correctly identified 869/960 changes at a rate of 12.9 per hour
Gunther et al. [34]	Respiratory variation (RV) monitoring device (RV monitor), using Strip chart and point and click method	AWP and ABP	Monitoring AWP and ABP continuously.	The study concluded that the point and Click Method is preferable.

Lowe [38]	SENTINEL monitoring system, that helps in FDD	HR, SAP, MAP, ETCO2, RR, BP, PV, SPO2.	Seven different physiological events	Sentinel has achieved sensitivity and specificity of above 90%
Lowe and Harrison [39]	Fuzzy logic based algorithm	SBP, HR and ETCO2	Malignant hyperpyrexia (MH)	System detected MH nine minutes before the anaesthetist diagnosed it
Esmaeilia et al. [40]	A fuzzy rule based system	Electroencephalogram (EEG) features	To estimate the depth of anaesthesia (DOA)	The mutual knowledge exchange between the human expert and the machine is simplified
Mahfouf et al. [41]	A Mamdani type of fuzzy model using ANFIS system	HR and SAP	Monitoring the changes in HR and SAP	The real SAP and the model result are 0.83 ($p<0:001$). The real HR and the model result is 0.86 ($p<0:001$)
Harrison and Connor [43]	An anaesthesia alarm system and further using Principal Component Analysis	SAP	Detects the changes in SAP	The system output is more clinically useful
Dosani et al. [44]	Alarm monitoring system using CUSUM	ETCO2, HR, MVex, NIAP, ABP, RR and Oxygen saturation	To detect the change in the physiological trend	System detected more than 60% clinically useful

Bressan et al. [47]	ASYS system for data acquisition and control using TCI	Syringe pumps for different drugs	Three syringe pumps, the BIS Monitor, and the AEP Monitor	System proved to be more efficient in providing shorter recovery time
James et al. [49]	An alarm system based on a modified proportional-derivative (PD) controller algorithm	Invasive arterial pressure	Hypotensive episodes	The false positive rate was 34% compared with 45–64% for the other alarms ($p < 0.01$).
Zong et al. [64]	Fuzzy logic based algorithm system	ABP and ECG	Reducing false alarms related to changes in arterial blood pressure (ABP).	System rejected 98.2%, reduces from 26.8% to 0.5% and accepted 99.8% of true ABP alarms.
Makivirta et al. [66]	An alarm system using single median filter and dual-limit alarm system (two median filters)	HR, ABP, PAP and CVP	Compared normal system with median filtered system	Single median filter: 81% true alarms detected and false alarms decreased by 66%
Koski et al. [71]	A trend detection algorithm (symbolization of online monitoring data)	HR, SAP, MAP, PAP CVP, CT and PT	Level symbolization and trend detection	Agreement with clinician: Level symbolization: 99.4% Trend detection: 93%

Charbonnier et al. [72]	An alarm system with Online-segmentation algorithm	SPO2	Comparison of threshold alarms and filtered signal alarms	19 true alarms and 6 technical alarms detected. 20 of the 24 false alarms were rejected.
Backer et al. [8]	Fuzzy logic intelligent alarm system	ABP and LAP	Evaluations of five different hemodynamic state variables	Achieved sensitivity: 99.3% Specificity: 66% Prototype: sensitivity: 99.07% specificity: 99.78%

1.5 Research Contribution

The original contribution of this thesis is as follows:

- Development of Smart Anaesthesia Monitoring System Using SCADA software

The system is developed for the condition of hypovolaemia in a patient under anaesthesia. The alarms were classified as mild, moderate, and severe. The developed system has been tested in over 50 hours of the data recorded from the actual operations. The system shows a high agreement (> 95% cases) with expert anaesthesiologists. On average, the system can predict hypovolaemia in a shorter time interval than the clinician.

- Development of SCADA techniques, conditions and GUI

In the development of the system, the built-in tools like real-time alarm charts, real-time graphical trend display and the Standard Deviation (SD) features in SCADA are studied and developed to generate warnings for hypovolaemia. The conditions were set in such a way that initially each parameter should satisfy a set threshold value; secondly, it should satisfy the change of SD setting (approximately 2–3 SD); and finally, all three parameters should satisfy the condition of hypovolaemia to generate the final warning or alert the clinician. In addition, the graphical user interface has been developed to assist the clinician in a more user friendly manner, and to present information in the simplest form so that it is easy to understand and use in a real time setup.

- Development of fuzzy logic based Smart Anaesthesia Monitoring System

The other major contribution to this thesis is the development of a fuzzy logic based system to detect the condition of hypovolaemia in the patient under anaesthesia. The alarms were classified as mild, moderate, and severe.

- Development and testing with different fuzzy modules

The new fuzzy modules in the development of the fuzzy based system are studied and implemented. Firstly, the data is tested under fuzzy (clustering); secondly, using Sugeno or Takagi-Sugeno-Kang type of fuzzy inference, the data is trained, and errors are reduced in the fuzzy IF-THEN rules and membership functions. Finally, a Mamdani-type inference is used to build the membership functions, logical operations, and IF-THEN rules.

- Comparison of both methods and results

After development and testing, both systems; i.e. SCADA SMS and fuzzy logic FLMS are compared (Chapter Five).

1.6 Structure of the Thesis

CHAPTER 1: Introduction

This chapter presents the background, motivations, literature review, and contributions of the research. The theoretical development of the research concepts and the diagnostic process are also discussed.

CHAPTER 2: Data Collection and Data Conversion

This chapter discusses the details of patients' data collection in the operating theatre, and then the conversion of the Digital files (.DOF) to a readable format (.txt) using a DOMonitor application.

CHAPTER 3: Pre-processing of the Patients' Data and Using SCADA Software for Anaesthesia Monitoring (SMS)

This chapter presents the pre-processing part of this work, where different techniques were adopted and tested to improve the results to achieve greater accuracy and offer more clinical importance. It includes the use of SCADA industrial software in the medical environment for the testing and diagnosis of different conditions in anaesthesia monitoring; additionally, the developments of conditions, rules, and GUI with the testing results are discussed.

CHAPTER 4: Fuzzy Logic and its Application to Hypovolaemia Detection (FLMS)

This chapter describes the development of the fuzzy logic based system with different anaesthetic conditions. Also discussed are different approaches used in the fuzzy logic modules, rules, and techniques.

CHAPTER 5: Overviews, Testing, and Results

This chapter presents the testing and results from both systems (SMS and FLMS).

CHAPTER 6: Discussion and Conclusions

This chapter compares the methods and results of two techniques (SMS and FLMS) and discusses the main conclusions with suggestions for future work.

1.7 Summary

The main aim of this thesis is to extend the work of Andrew Lowe, Michael Harrison, Hamid GholamHosseini, and Bhupendra Gohil [32, 38, 39, 73-75] by developing real time algorithms for analysis of the hemodynamic data from the patient and providing a warning/alarm to the anaesthetist. The developments of the alarm system are

discussed in the following chapters. It includes the newly developed SCADA alarm algorithms, SCADA testing rules and conditions, and the system is called SCADA Monitoring System (SMS). Also, using the fuzzy logic technique, Fuzzy Logic Monitoring System (FLMS) is developed and its rules and diagnostic alarm algorithms are discussed.

CHAPTER 2 Data Collection and Conversion

2.1 Introduction

The managerial aspect of providing health services to patients in hospitals is becoming increasingly important. Hospitals want to reduce costs and improve their financial assets on the one hand, while they seek to maximize the level of patient satisfaction on the other. One unit that is of particular interest is the operating theatre (OT). Since this facility is the hospital's largest cost and revenue centre, it has a major impact on the overall performance of the hospital [76, 77]. Moreover, health professionals have to anticipate the increasing demand for surgical services caused by the ageing population [78]. These factors clearly emphasise the need for efficiency, and the necessity for further development of the operating theatres. The past few decades have witnessed a real improvement in the operating theatre setup and patient monitoring equipment; this benefits both the clinician and the patient. Even patient monitoring while they are being transported to the hospital provides data on such vital and complex parameters as electrocardiography (ECG), oxygen saturation by pulse oximetry (SpO_2), heart rate, and blood pressure [79] helps with early treatment and makes the clinician's work easier.

One of the important areas related to this project is the patient's anaesthesia record.

The anaesthesia record is an essential part of the patient's medical record. Even the best of anaesthetic care cannot be defended or referred to if there is no clear record that such care took place. This applies to pre-operative evaluation and postoperative care, as well as intra-operative management. The essential purpose of maintaining the anaesthesia record is to document how an individual patient responds to anaesthesia and surgery. Giving anaesthesia to a patient is a *process;* it follows a logical sequence. This includes a pre-anaesthetic checkup, medical optimization, planning of the anaesthetic technique, administration of anaesthesia, and postoperative management. The findings and interventions undertaken during this process are documented in what constitutes an *anaesthesia record*.

The anaesthesia record is a generic document that is used for a wide variety of patients and procedures. An ideal record system is expected to contain relevant patient

information: procedure, pre-anaesthetic evaluation, anaesthesia technique details, intra-operative events/complications, and post operative instructions of the anaesthesiologist. This record can be, and has been, used for offline analysis, fault detection, development of monitoring systems, development of anaesthesia warning systems, and also in a number of different areas of studies related to it. In the following sections of this chapter, the details of patients' data collection are discussed, followed by data conversion techniques and steps.

2.2 Experimental Setup

The rapid development of mobile technologies, including increased communication bandwidth and miniaturization of mobile terminals, has accelerated development in the field of mobile telemedicine [80]. Hamid et al. [81] have done work towards the development of remote monitoring systems which offer the potential to assist anaesthesiologists and clinicians during surgery. This work also explores the possibility of realizing a reliable and cost-effective remote monitoring system and the development of a decision-support system to function in a smart alarm capacity and counter the complexity of modern anaesthetic procedures. Wireless patient monitoring systems in the OT not only increase the mobility of patients and medical personnel, but also improve the quality of health care [82]. With respect to the remote monitoring of patients, many groups have demonstrated the transmission of vital biosignals using global systems for mobile communication (GSM) technology [83, 84]. Some researchers have used cellular phones to transmit vital signs from the ambulance to the hospital, either in store-and-forward mode [85] or in real-time mode[86]. In the opinion of this project, the figures below show the practical setup for the real-time test bed during a real-time data collection and testing session at the Auckland City Hospital. The goal of the real-time data collection was to capture the anaesthetists' diagnoses and the patient data with the correct time-stamp, and to evoke suggestions from the anaesthetist for making the prototype alarm more ergonomic.

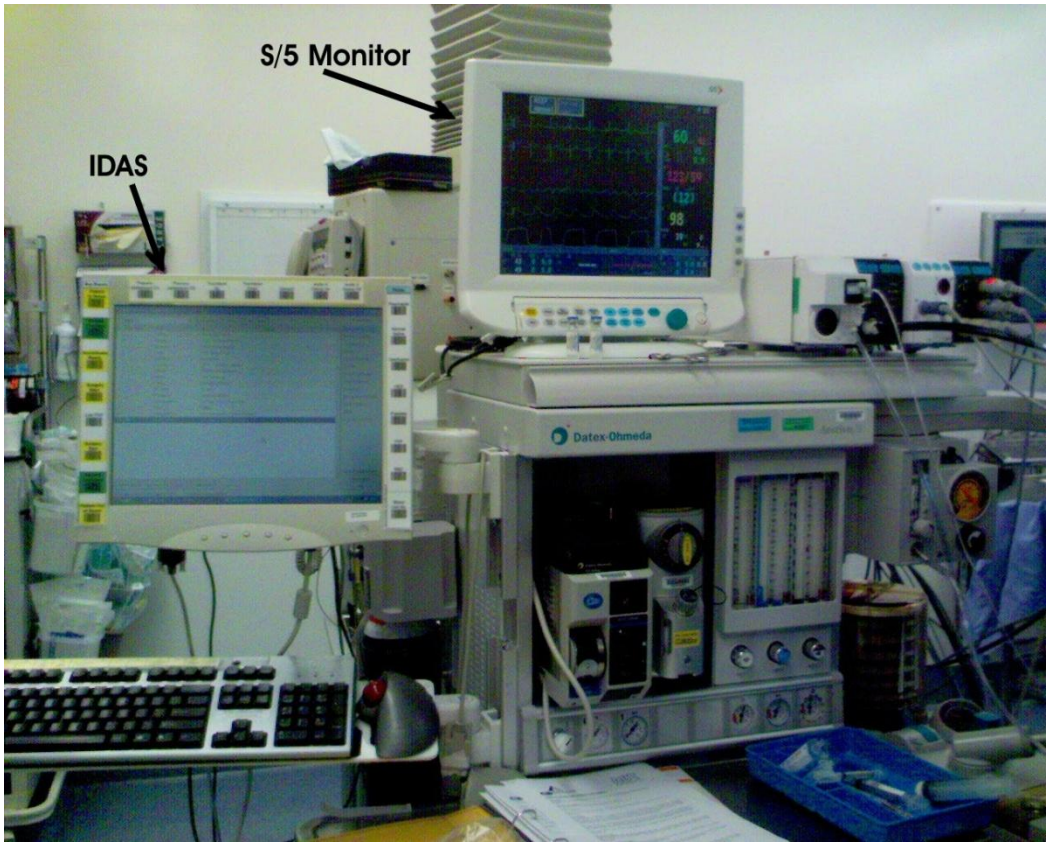


Figure 1.4.8-1. Hospital's operating theatre setup.

2.3 Ethics Approval and Patient Consent

Physiological data collection from the patient monitor requires ethical approval from the local ethics committees; these approvals were obtained from the Northern-X Regional Ethics Committee and from the Auckland University of Technology Ethics Committee (AUTEC). As part of the ethics approval, an informed written consent was required from each patient who participated in the research. Before signing the consent form, each patient was informed about the project by the anaesthetist and the participant was given a patient information sheet so he/she would understand the purpose of the research. The data were collected by an anaesthetist in the Auckland City Hospital operating theatre suite from the patients undergoing major surgery.

2.4 Data Collection

One of the many tasks that occupy anaesthetists during an operation is documentation — the anaesthetic record. This includes a log of various physiological parameters, drugs used, blood lost, fluids administered, procedures performed, etc. Since this is generally a hand-written record, the documentation task can become rather neglected

during busy periods; consequently, anaesthetists are increasingly using computers to automate the collection of such data. This has many advantages, including the real-time processing of data, generation of various derived parameters, and greatly enhanced information display facilities. The physiological data were directly collected from the S/5 Datex-Ohmeda (GE, Datex-Ohmeda, Helsinki, Finland) anaesthesia monitor. A large amount of patient data used for offline (retrospective) analyses were collected by the project's external supervisor, Dr. Michael Harrison. These data were collected from patients in the United Kingdom and New Zealand, with the respective local ethical approvals obtained. These data were collected using a software program called S/5 Collect from GE Healthcare Limited. S/5 collect, however, supports only data-logging to a digital file and does not relay the data to any other device or application. Therefore S/5 collect cannot be used for real-time testing.

Real-time data-collection involves tracking and monitoring patients for data-collection and setting up a desktop computer running in an operating theatre prior to every data-collection session. The patients recruited for real-time data collection were patients who were likely to suffer moderate to major blood loss due to the nature of the surgery. Another important criterion to be considered during patient recruitment was that only those patients who needed an arterial pressure line (for measuring BP/P1 signal invasively) during the surgery could be recruited for data collection.

2.4.1 Data Collection using Datex-Ohmeda S/5 Monitor

Figure 2.4.1-1 shows a screen-Short of the Datex-Ohmeda S/5 monitor during a real-time procedure.

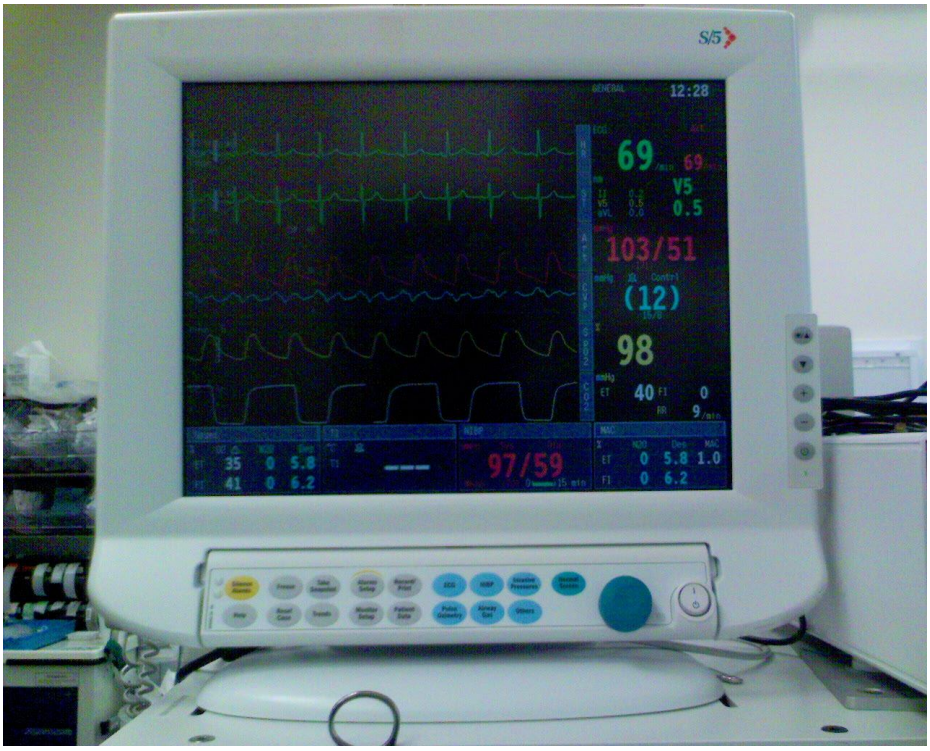


Figure 2.4.1-1. Screen-shot of the Datex-Ohmeda S/5 monitor during a procedure.

2.4.2 Data Collection using Anaesthesia Monitor

The Datex-Ohmeda S/5 anaesthesia monitors have a high-speed asynchronous serial interface for data acquisition purposes. The interface provides access to the physiological database of the monitor, which contains the most important measurement values and associated status information. Since this monitoring equipment is used in critical care environments, it has an RS-232 serial interface for the process of data collection, construction of trend graphics, and formatting and typesetting have been automated to be used with reasonable ease.

2.4.2.1 Line Parameters and Line Structure

The interface uses the following serial communication line parameters: 19,200 bit/s transmission rate with eight data bits with even parity and one stop bit. The CTS/RTS hardware hand-shaking is used for communication control. All data to and from the monitor is transferred using flag-delimited frames. Each data frame starts and ends with a flag character. All application data is always located between these flags.

2.4.2.2 Data Structure

All data is in binary '80x86' format with 32 bits. The lower byte of a 16-bit word is saved in the lower address and its higher 16-bit word is saved in the higher address. The lower word of a 32-bit value is saved in the lower address and its higher 32-bit word is saved in the higher address. Sizes of integer and short integer are eight bits, 16 bits and 32 bits respectively. Some commonly used data types are defined:

define byte	unsigned char	8-bits
define word	unsigned short	16-bits
define dword	unsigned long	32-bits

2.4.2.3 Physiological Data Structure

There are various physiological signals available as output from the S/5 anaesthesia monitor. The signals of interest to us were heart rate (HR), blood pressure (BP), pulse volume (PV) and end tidal CO_2 (ETCO2). These signals represent the analogue waveform information of the signals in a digital time-series format. Digital signals are derived by sampling the analogue signals coming from various sensors at a particular sampling rate. For instance the analogue signal coming from the pulse oximetery probe is sampled at a frequency of 100 Hz; i.e., 100 digital samples are recorded in a buffer at one-second intervals. Since the digital representation of the analogue signal is discrete and not continuous in time, these digital signals are also called *discrete-time series*.

The raw signals acquired from the S/5 anaesthesia monitor have embedded noise/artefacts due to various pneumatic, hydraulic, optical, and electronic devices. These noise/artefacts need to be discarded. Pre-processing of the signals involves filtering and smoothing of the original signal to make it noise free. A number of signal processing techniques were adopted for filtering the signals; this is discussed in the next chapter.

2.4.2.4 ECG-Heart Rate Structure

Every physiological parameter inside the anaesthesia monitor is positioned with the related groups and sources, and also assigned to each separate bit for each source. Below is the c-code for the collection of ECG-Heart Rate from the patient to the anaesthesia monitor.

```
struct ecg_group

{
struct group_hdr hdr;
short hr;
short st1;
short st2;
short st3;
short imp_rr;
};
```

In the above c-code, the defined group is called a *header* for the ecg_group set to run as a collection for the ECG data; similar codes are assigned for all the parameters, related to their groups and sources.

2.4.2.5 Waveform Data Structure

The Datex-Ohmeda monitor interface provides access to the real-time waveform data produced by the monitor. Accessing the waveforms does not exclude access to the physiological data. As a significant data transmission rate is needed to produce a real-time waveform, this interface supports waveform transmission only up to a total of 600 samples (1200 bytes) per second.

The following c-code shows how fields are used as the waveform is requested in real-time and processed by the monitor.

```
struct wf_req
{
short req_type; // request type
short secs; // duration of snapshot
byte type[DRI_MAX_SUBRECS]; // waveform selectors
byte addl_type[2*DRI_MAX_SUBRECS]; // waveform selectors
short reserved[10]; //
};
```

1. “req_type” is the waveform request specifier.
2. “Reserved” is reserved for future use.
3. “Type” is an array of the requested waveform sub records.

There is room for up to eight waveforms, but the monitor sends only the waveforms that fit within the 600 samples limitation and ignores the rest. Each waveform record contains one or more sub-records, the types of which match the types specified in the waveform request. Depending on the total number of samples, the monitor sends a waveform packet every 1000 ms, 500 ms or 250 ms. Waveform sub-records are of variable length. The actual length of the sub-record is included in the sub-record itself.

2.4.3 Data Collection using S/5 Collect Software

S/5 Collect is data collection software from GE Healthcare (the manufacturer of S/5 Datex Ohmeda anaesthesia monitor). However, the S/5 Collect software application can only collect data from the S/5 monitor into the data collection computer; the acquired data can then be saved into a digital data file in DOF format which can then be used for offline analysis. It has no provision for relaying data to any other device or application; therefore, it cannot be used for real time data collection and testing. In the past, S/5 Collect was used to collect data from the anaesthesia monitor, but then the serial port was not occupied by IDAS; this data was collected for offline testing only. Some of the data which was collected using S/5 Collect in the past was used for offline testing of this project.

2.4.4 Digital File Format

The computer labelled ‘IDAS’ is a data logging computer used by the Auckland City Hospital’s computerized database. The ‘S/5 monitor’ continuously relays the waveform data and the physiological data through a single serial port at the back of the S/5 monitor. The IDAS system acquires the patient data from the S/5 monitor via the serial port and saves it into the patient record as a DOF file format with *.dof* as the file extension. To accomplish this serial communication between S/5 and IDAS,

various handshake signals are continuously exchanged between the S/5 monitor and the IDAS system [87].

The conversion of a file from .dof to readable format is one of the main goals of this research project, because the .dof format file cannot be opened with a general application, and the main physiological data content of the patients cannot be seen in real-time or offline mode. There was an immediate need for appropriate software, or an application that could convert .dof format into normal readable format (i.e., .txt, .csv, .xls) to proceed with the offline work. The following section discusses the file conversion utility using the DOMonitor application.

2.5 Data Conversion

2.5.1 DOMonitor.Net Application

DOMonitor.Net is a JAVA .NET-based data collection application developed by Dr. Andrew Lowe, (Pulsecor Limited). Originally DOMonitor was used to acquire data from the S/5 monitor, save captured data to a digital file and simultaneously relay the data over another serial port. The digital file saved by the application can be used for offline analysis. DOMonitor had to be modified so that the acquired data could be relayed for real-time analysis. This application served as a very handy tool for testing as it performed the tasks simultaneously; this streamlined the whole process. It acquires data from the S/5 monitor and relays it to IDAS over another serial port, and also transmits the required data signals over a Transmission Control Protocol (TCP) port. It saves the selected waveform data to a readable digital file that can be accessed in offline mode for retrospective analysis.

2.5.2 Hardware and Software Requirements

Hardware required by the DOMonitor application is a standard computer, with the addition of serial ports to communicate. These serial ports link with other machines for conversion and transfer of data. The data collection card is for capture of data from different machines.

S**oftware** required is Microsoft Visual Studio 2005. The DOMonitor application is built into the dot net platform, called DOMonitor.net which runs in Visual Studio 2005.

Reference Libraries: To build and run this project successfully, the reference library nplot.dll was added to the project build; and the dot net framework was used as the supporting file.

2.5.3 How DOMonitor.Net Application Communicates

This application takes the data in the packet format from the .dof file and converts it into frame by frame format with 10s intervals and continuous relay. The converted data and the waveform signal are transferred to the user-created empty file; date and time are also recorded simultaneously as shown in the Figure 2.5.3-1.

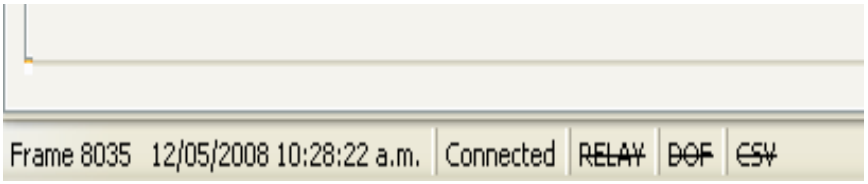


Figure 2.5.3-1. DOMonitor.net.

2.5.4 Steps for Data Conversion using DOMonitor.Net Application

There are some simple steps to use this application:

1. Open Microsoft Visual Studio 2005.
2. Load the main project file: DOMonitor.vdproj.
3. Add the reference library NPLOT: nplot.dll.
    a. Build project from **build**
    b. Run project from **run** tab
4. Now the project is in running mode.

2.5.4.1 Errors

The DOMonitor application needs to send the data packet into the main server with the assembly file and then recall it from the server. This needs to be done because of

the frame by frame conversion; for this, the administration login is required, otherwise it will show an error message: *the assembly file missing or not located in the computer*.

After the project is successfully built, it is ready to run. Press **run** button; the DOMonitor.Net window will appear on the screen. From this point onwards, the procedure for the file conversion is the same as with the executable version, which is described in the next section of this chapter.

2.5.5 Steps for Data Conversion using DOMonitor.Net Executable Version

The latest and modified version of the DOMonitor.net application makes the conversion work easier and smoother. By using this executable version, the data conversion can be done without Microsoft Visual Studio software. This application can be run easily in any computer without change in the main source code of the program.

The simple steps for data conversion are as follows:

- Double click on the icon named *DOMonitorNet.exe*
- A DOMonitor.net window will appear — see Figure 2.5.5-2.

There are four tabs in the window: Connect, Data, Log, and About as explained below see Figure 2.5.5-1:

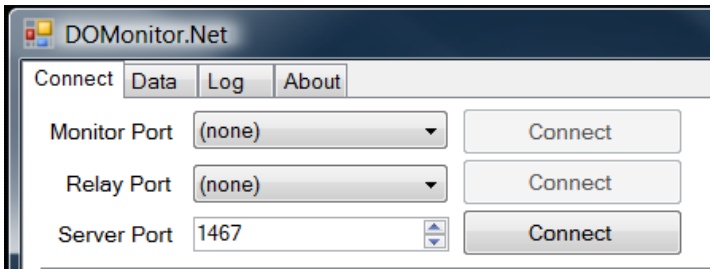


Figure 2.5.5-1. DOMonitor.net window containing four tabs.

Connect Tab

1. It contains Monitor port with three options.

    - COM1 and COM3 –use to upload data from communication ports.
    - FILE – use to load data from a digital file.
    - Figure 2.5.5-2 shows three options from the monitor port dropdown.

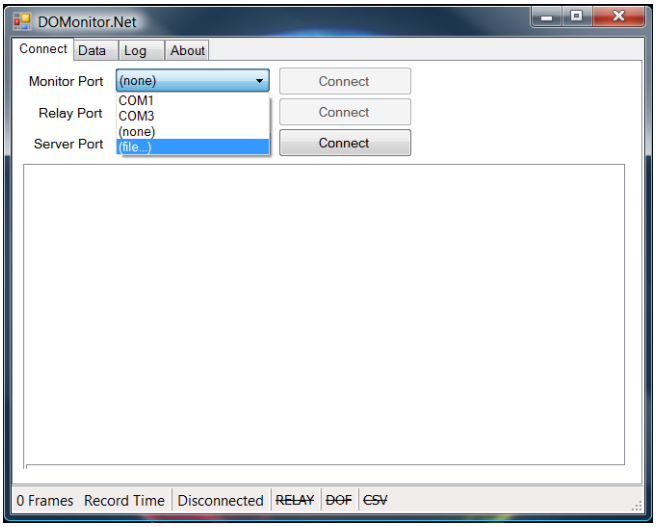


Figure 2.5.5-2. DOMonitor drop down dialogue box.

2. Relay Port – This is used to connect to the online port for continuous relay of data from the communication port.
3. Server Port – This is used to connect directly to the server with the following steps:

- Select **File** in the Monitor Port of Connect tab.
- Select the .dof file from the hard drive location.

- In this example, patient 30.dof is taken see Figure 2.5.5-3.

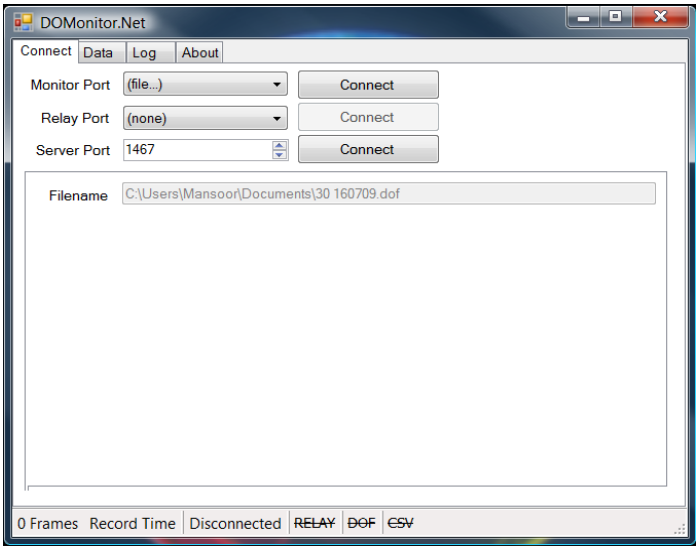


Figure 2.5.5-3. Connect Tab in DOMonitor.net.

Log Tab

This has two options: Save DOF File and Save Text File.

1. Save DOF File – It will save the converted file as DOF file.

2. Save Text File – It will save the converted file as TEXT file.

3. Select **Save Text File** — It will ask user to create an empty text file in the hard drive for the conversion to take place.

In this example, Patient-30.csv is created using the following steps see Figure 2.5.5-4:

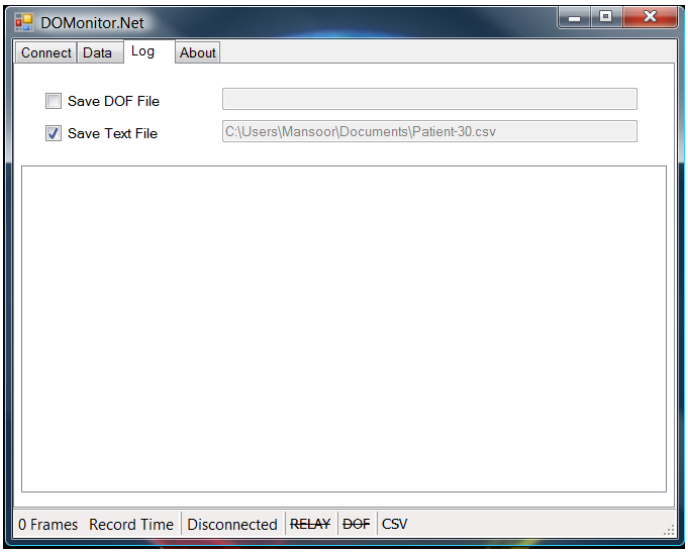


Figure 2.5.5-4. Log Tab in DOMonitor.net.

- Press **Connect** button from the connect tab and the data conversion starts:
    - Converting frame by frame with date and time. See Figure 2.5.5-5.
    - Converting the .dof file and save it as text file like .txt or .csv or .xls. As shown in Figure 2.5.5-5.

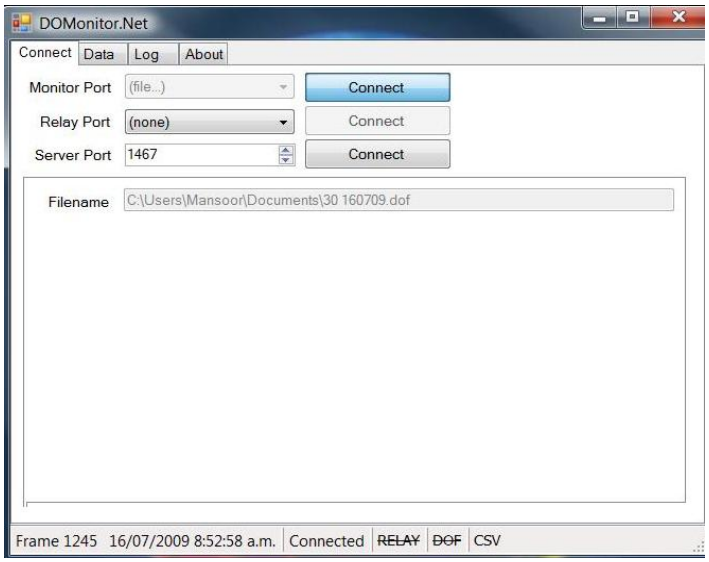


Figure 2.5.5-5. Working of DOMonitor.net Application.

Data Tab

This tab contains all the trend and waveform data of the patient's important physiological features; this can be used for the analysis. Figure 2.5.5-6 shows the waveform signal of Invasive Pressure (INVP1) in real-time.

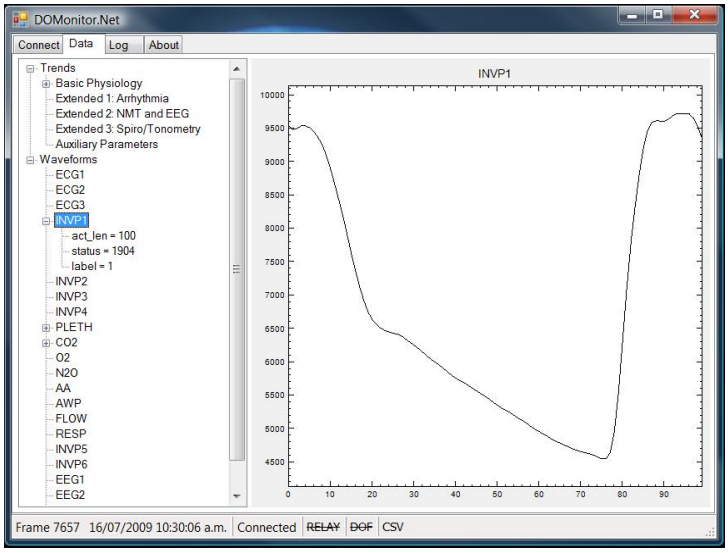


Figure 2.5.5-6. Waveform of Invasive Pressure (P1).

While the conversion of the file is still running, the waveforms and trends can be seen and saved simultaneously. Figure 2.5.5-7 shows the waveform of the physiological feature Carbon Dioxide (CO_2).

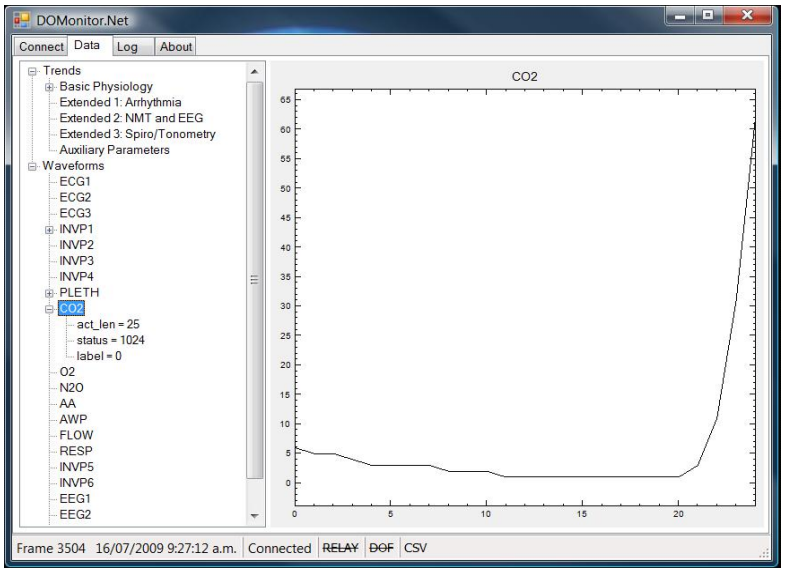


Figure 2.5.5-7. Waveform of Carbon Dioxide (CO2).

2.5.6 Converted Text File

After completion of the application, the empty created file is stored with text format data which is used for offline analysis in this project work. Figure 2.5.6-1 shows the first 29 rows and first 13 columns of the total patient's data file (saved in excel format).

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	rtime	ecg.hr	ecg.st1	ecg.st2	ecg.st3	ecg.imp_rr	p1.sys	p1.dia	p1.mean	p1.hr	p2.sys	p2.dia	p2.mean
2	16/07/2009 8:34	-32766	-32767	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
3	16/07/2009 8:34	-32767	-32767	-32767	-32767	-32767	-32764	-32764	-32764	-32764	-32764	-32764	-32764
4	16/07/2009 8:34	80	-32767	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
5	16/07/2009 8:34	78	-32767	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
6	16/07/2009 8:34	82	-32767	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
7	16/07/2009 8:35	79	-32767	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
8	16/07/2009 8:35	77	0	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
9	16/07/2009 8:35	78	-135	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
10	16/07/2009 8:35	78	-194	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
11	16/07/2009 8:35	78	-109	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
12	16/07/2009 8:35	79	-242	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
13	16/07/2009 8:36	78	-149	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
14	16/07/2009 8:36	78	-142	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
15	16/07/2009 8:36	78	-130	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
16	16/07/2009 8:36	79	-156	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
17	16/07/2009 8:36	77	-117	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
18	16/07/2009 8:36	76	-149	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
19	16/07/2009 8:37	76	-156	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
20	16/07/2009 8:37	78	-153	-32767	-32767	-32767	-32764	-32764	-32764	0	-32763	-32763	-32763
21	16/07/2009 8:37	77	-171	-32767	-32767	-32767	-32763	-32763	-32763	0	-32763	-32763	-32763
22	16/07/2009 8:37	79	-161	-32767	-32767	-32767	-32763	-32763	-32763	0	-32763	-32763	-32763
23	16/07/2009 8:37	82	-149	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
24	16/07/2009 8:37	84	-102	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
25	16/07/2009 8:38	84	-102	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
26	16/07/2009 8:38	82	-104	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
27	16/07/2009 8:38	79	-82	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
28	16/07/2009 8:38	75	-100	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764
29	16/07/2009 8:38	72	-135	-32767	-32767	-32767	-32764	-32764	-32764	0	-32764	-32764	-32764

Figure 2.5.6-1. Patient's data file.

The total file of each patient contains approximately 60 of their physiological features, and the length of the file depends upon the duration of operation (number of hours) as the rows reflect a time interval of 10 seconds. It can be saved as a .xls format, .csv format, or .txt format file.

2.6 Plotting the Data Using Matlab

From the patient's readable data files, ECG-heart rate of Patient-30 is taken to plot the complete waveform signal using Matlab as shown in Figure 2.5.6-1.

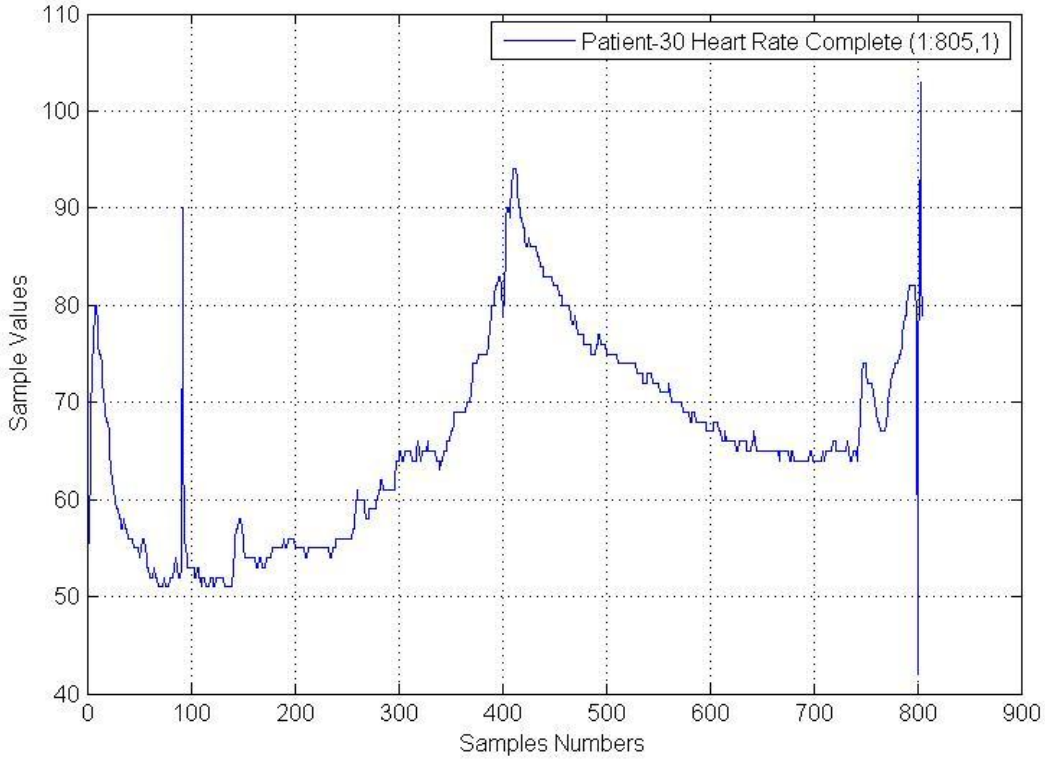


Figure 2.5.6-1. Waveform signal of patient #30 Using MATLAB.

2.7 Summary

The conversion of the patient's data into readable format has been successfully done by the DOMonitor.Net application program. The executable version is the latest modified version of this application and uses the frame-by-frame conversion technique with a 10-second time interval. The important physiological data can now be saved as readable format. By the use of this application, analysis of offline data is carried out easily throughout this project work.

The next part is data analysis using different techniques and methods to test the systems. In this analysis the first step is focused on the removal of noise and artefacts from the collected patient's data. The details of data analysis are discussed in the next chapter.

CHAPTER 3 Pre-processing of the Patients' Data and Using SCADA Software for Anaesthesia Monitoring

3.1 Introduction

This section outlines the pre-processing of the patients' data for further analysis; a brief outline of SCADA system as well as the use of SCADA software for the anaesthesia monitoring SMS system is given later.The next section describes the world of Digital Signal Processing (DSP) by describing the dramatic effect that DSP has had using different techniques and methods for the pre-processing of data used for the alarm system examined in this project. It also covers the DSP techniques and data analysis methods, which is followed by a brief explanation of SCADA and its software development.

3.2 Pre-processing of the patients' data

Digital signal processing is one of the most powerful technologies shaping science and engineering in the twenty-first century. Revolutionary changes have already been made in a broad range of fields such as communications, medical imaging and monitoring, radar and sonar, high fidelity music reproduction, and oil prospecting. Each of these areas has employed a deep DSP technology with its own algorithms, mathematics, and specialised techniques. Digital signal processing is distinguishable from other areas in computer science by the unique type of data it uses – signals. In most cases, these signals originate as sensory data from the real world such as seismic vibrations, visual images, and sound waves. DSP is the mathematics, the algorithms, and the techniques used to manipulate these signals after they have been converted into a digital format. This system includes a wide variety of goals such as the enhancement of visual images, recognition and generation of speech, compression of data for storage and transmission, and the extraction of physiological features.

The raw signals acquired from the anaesthesia monitor have embedded noise/artifacts due to various pneumatic, hydraulic, optical, and electronic devices. These noise/artifacts need to be discarded; this is the pre-processing and involves filtering and smoothing of the original signal. The following section describes the development for each of the filtering techniques employed in this system.

3.2.1 Analysis of the Raw Signals

A signal is a pattern of variation of a physical quantity: a definition which covers a wide territory. Signals are all around us. Examples include acoustical, electrical, and mechanical signals. Signals may depend on one or more independent variables.

Signals can be categorized as analog, discrete, or digital. They are summarized as follows:

- **Analog** signals: These vary continuously in amplitude and time. The independent variable is not necessarily time; it could also be a spatial coordinate.
- **Discrete-time** signals: signals that have continuous amplitude but only exist at discrete times. These signals are represented as sequences of numbers.
- **Digital** signals: signals that have discrete amplitude and time. These signals are represented by sequences of numbers with finite precision. They are used when processing information by computer.

Noise/artifacts in the raw physiological signals have different frequency components, and thus cannot be eliminated by employing a single band stop filter. Various filtering techniques with different specifications were tried in order to achieve the desired level of filtering. The cleaned signal waveforms obtained through noise filtering were to be used for diagnosis.

3.2.2 Spectral Analysis

In order to define filter specification for a signal, it is essential to distinguish between the useful frequencies and the noisy frequencies in the raw signals obtained from the anaesthesia monitor. This process of analysing the frequency content in the signal is called spectral analysis. To obtain the frequency content of a discrete time signal we need to transform the time domain signal into the frequency domain. The DSP tool used for transforming the time domain signal into the frequency domain signal is called Discrete-Time Fourier Transform (DTFT). The physiological signals have specific frequency components which represent the rate of changes of the physiological variables. The frequency spectrum of the signals can be used for distinguishing the clean signal samples from the noisy samples [88].

3.2.3 Discrete-Time Signals

In digital signal processing, signals are represented as sequences of numbers called samples. A sample value of a typical discrete-time signal or sequence is denoted as *x[n]* where the argument *n* is an integer between $-\infty$ and ∞. It should be noted that *x[n]* is defined only for integer values of *n* and is undefined for non-integer values of the argument *n* [88].

The Fourier transform has long been used for characterizing linear systems and for identifying the frequency components making up a continuous waveform. However, when the waveform is sampled, or the system is to be analyzed on a digital computer, it is the finite, discrete version of the Fourier transform (DFT) that must be understood and used. Although most of the properties of the continuous Fourier transform (CFT) are retained, several differences result from the constraint that the DFT must operate with on sampled waveforms defined over finite intervals.

'The Fast Fourier Transform (FFT) is simply an efficient method for computing the DFT...' The Fourier Transform´s ability to represent time-domain data in the frequency domain and vice-versa has many applications. One of the most frequent applications is analysing the spectral (frequency) energy contained in data that has been sampled at evenly-spaced time intervals. Other applications include fast computation [89].

3.2.4 Discrete-Time Fourier Transform (DTFT)

The discrete-time Fourier transform (DTFT) of a discrete-time sequence *x[n]* is a complex exponential representation $\{e^{-j\omega n}\}$ where ω is a real frequency variable. The DTFT $X(e^{j\omega})$ of a discrete-time sequence *x[n]* is defined by Eq.3.1.

$$X(e^{jw}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \qquad \textbf{3.1}$$

3.2.5 Discrete Fourier Transform (DFT)

For a finite-length sequence *x[n]*, $0 \leq n \leq N-1$ there exists a simpler relationship between the sequence and its DTFT $X(e^{j\omega})$. For a length N sequence, only N values of $X(e^{j\omega})$, called the frequency samples, at N distinct frequency points, $\omega = \omega_k$, $0 \leq n \leq N-1$, are sufficient to determine *x[n]*, and hence, $X(e^{j\omega})$; this frequency domain

representation of a finite-length sequence is called discrete Fourier transform (DFT). The DFT is one of the most commonly used DSP tools for analysing the frequency content of a time-series signal. By definition the DFT of a sequence *x[n]* is given by following Eq.3.2.

$$X[k] = X(e^{j\omega})\Big|_{\omega=2\pi k/N} = \sum_{n=0}^{N-1} x[n]e^{-j2\pi\frac{k}{N}n} \qquad 0 \le n \le N-1 \qquad \textbf{3.2}$$

The DFT of a sequence can be computed efficiently in practice using a fast Fourier transform (FFT) algorithm. MATLAB provides built-in functions for the computation of the DFT. The power spectrum of the signal gives the measurement of power in a signal at different frequencies. These functions were implemented during the data analysis for identifying the frequency spectrum of the clean physiological data in the raw signals. After performing the DFT of the BP and PV signals a power spectrum of the signals was plotted. The clean signals are identified by the high power band in the frequency spectrum. Figure 3.2.5-1 shows the power spectrum of the BP signal transformed to the frequency domain using FFT algorithm. It illustrates that the frequency of clean BP data is in the range 0 Hz to 2 Hz (high power content).

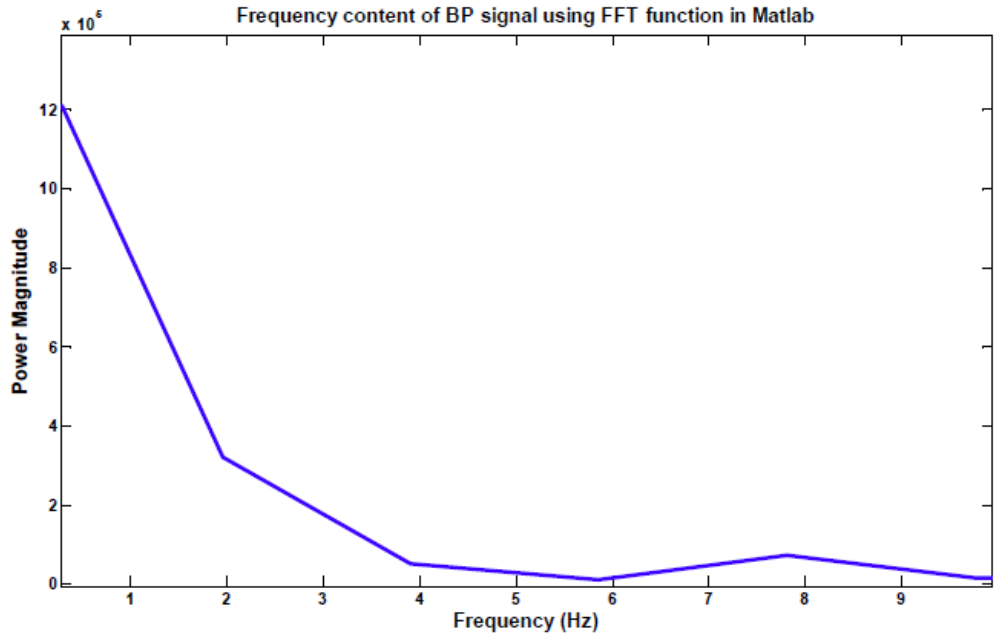


Figure 3.2.5-1. Power spectrum of BP signal.

3.2.6 Filtering of the raw signals

Each filter functions by accepting an input signal, blocking prespecified frequency components, and passing the original signal, minus those components to the output. For example, a typical phone line acts as a filter that limits frequencies to a range considerably smaller than the range of frequencies human beings can hear.

In a typical digital filtering application, software running on a digital signal processor (DSP) reads input samples from an A/D converter, performs the mathematical manipulations dictated by theory for the required filter type, and outputs the result via a D/A converter.

There are many filter types, but the most common are lowpass, highpass, bandpass, and bandstop.

3.2.7 Frequency response

Simple filters are usually defined by their responses to the individual frequency components that constitute the input signal. There are three different types of responses. A filter's response to different frequencies is characterized as passband, transition band, or stopband. The passband response is the filter's effect on frequency components that are passed through (mostly) unchanged.

Frequencies within a filter's stopband are, by contrast, highly attenuated. The transition band represents frequencies in the middle, which may receive some attenuation, but are not removed completely from the output signal.

3.2.8 Finite Impulse Response

A finite impulse response (FIR) filter is a filter structure that can be used to implement almost any sort of frequency response digitally. An FIR filter is usually implemented by using a series of delays, multipliers, and adders to create the filter's output.

3.2.9 Lowpass Filter

A lowpass filter passes the signals in the lower spectrum of the frequency and blocks the signals in the higher frequency range. Some signal sources generate

interference/noisy signals and these sources are ambient sources producing noisy signals of much higher frequency than the useful signals. For instance, noise due to a cauterization procedure usually has a frequency much higher than the signal under consideration. Other artifacts are embedded in the signal from the sensor itself. For example, touching or moving the invasive pressure sensor will generate noise in the P1 signal, which is highly contaminated. By filtering the signal with a lowpass filter noise from high frequency, noise sources can be eliminated. For in-depth theory on filter design refer to [88].

MATLAB™ has built-in filter modules that generate filter coefficients automatically for user specified filter requirements. Once the frequency range is identified for the filter the MATLAB FDAtool can be used for generating appropriate filter coefficientsFigure 3.2.9-1 shows a standard low pass filter specification and illustrates the symbols used in this context.

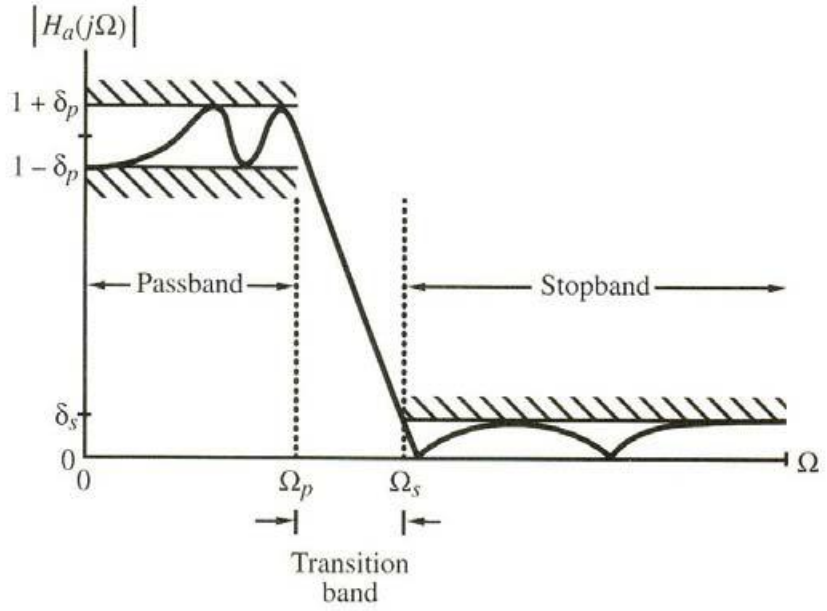


Figure 3.2.9-1. Low Pass Filter Specification.

Where,

Ω_p ➔ Passband edge frequency.

Ω_s ➔ Stopband edge frequency.

δ_p ➔ Passband ripple.

δ_s ➔ Stopband ripple.

$|H_a(j\Omega)|$ ➔ Magnitude of the filter.

All signals having frequencies in the range of 0 to Ω_p pass through the filter without any attenuation while the signals with frequencies greater then Ω_p are highly attenuated. Ideally a low pass filter should have a transition band of infinitesimally small bandwidth (i.e. $\Omega_s \approx \Omega_p$), which requires a very high filter order and thus increases the computation requirement for executing the filter algorithm. In practice the filter has a transition band of moderate width which is a compromise between the ideal filter specification and the available computational resources. Also the magnitude response of the passband and stopband is not constant, and specified with some tolerance limits in terms of the ripple content. Figure 3.2.9-2 shows the practical magnitude response of a lowpass filter for invasive blood pressure (P1) waveform. For in-depth theory on filter design refer to [88].

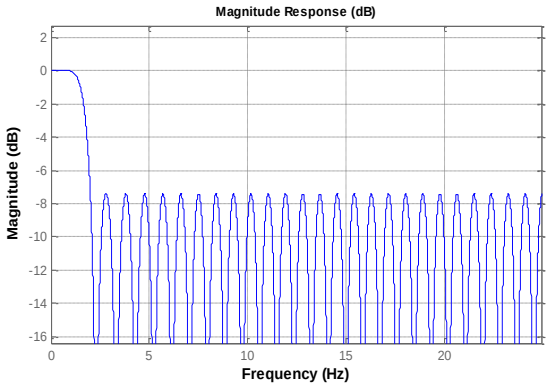


Figure 3.2.9-2. Practical Magnitude Response of a Lowpass Filter.

3.2.10 Adaptive Filtering

Adaptive filters are basically digital filters with self-adjusting characteristics (specifications). An adaptive filter consists of two distinct parts: a digital filter with

adjustable coefficients, and an adaptive algorithm which modifies the coefficients of the filter depending upon the noise characteristics of the contaminated signal. An adaptive filter was also tested with the sample of BP and PV data, but successful filtering could not be achieved due to the high variability of the noise in the input signals [87].

3.2.11 Variance Based Filtering

Variance and standard deviation are extensively used in the following algorithms and it is worthwhile to take a brief look at these terms. In statistics theory, the variance is defined as the measure of statistical dispersion of a random variable. Variance of a random number indicates how dispersed the population sample is for the random variable in the given data samples. The square root of the variance is called standard deviation [90]. Standard deviation measures how widely spread the values in a data set are. If the data points are all close to the arithmetic mean, then the standard deviation is close to zero. If data points are spread far from the mean, then the standard deviation is far from zero.

The variance based filtering technique discussed in this chapter was used for eliminating major artifacts. The variance based filtering algorithm analyses variance of data-batches, which represent a segment of the whole time series data, and filters the time-series data on a batch-by-batch basis [87].

After the pre-processing of the raw data which includes DSP methods and different filtering techniques, we used the SCADA system and SCADA software for the proposed anaesthesia monitoring system. This is described in the next chapter.

3.3 Supervisory Control and Data Acquisition (SCADA) System

3.3.1 Introduction

SCADA systems are used to monitor a setup and control a plant or equipment in industries such as telecommunications, water and waste control, energy, oil and gas refining and transportation. These systems encompass the transfer of data between a central host computer and a number of Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs), and the central host and the operator terminals (Figure 3.3.1-1 shows a typical SCADA system). A SCADA system gathers

information (such as where a leak on a pipeline has occurred), transfers the information back to a central site, then alerts the home station that a leak has occurred, carrying out necessary analysis and control, such as determining if the leak is critical, and displaying the information in a logical and organized fashion.

These systems can be relatively simple, such as one that monitors environmental conditions of a small office building, or very complex, such as a system that monitors all the activity in a nuclear power plant. Traditionally, SCADA systems have made use of the Public Switched Network (PSN) for monitoring purposes. Today many systems are monitored using the infrastructure of the corporate Local Area Network (LAN)/Wide Area Network (WAN). Wireless technologies are now being widely deployed for purposes of monitoring [91]. The key performance of a SCADA system can be summarised as:

- Data acquisition
- Networked data communication
- Data presentation
- Monitoring
- Control

SCADA is not a specific technology, but a type of application that gets data about a system in order to control that system.

A SCADA application has two elements:

- The process/system/machinery user want to monitor or control.
- A network of intelligent devices that interface with the system through sensors and control outputs.

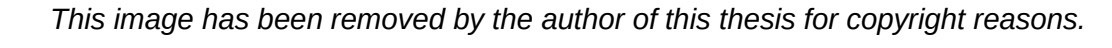


Figure 3.3.1-1. Typical SCADA system [92].

3.3.1.1 Field Data Interface Devices

Field data interface devices form the ‘eyes and ears’ of a SCADA system. Devices such as reservoir level meters, water flow meters, valve position transmitters, temperature transmitters, power consumption meters, sensors and pressure meters, all provide information that can tell an experienced operator how well a water distribution system is performing. In addition, equipment such as electric valve actuators, motor control switchboards, and electronic chemical dosing facilities can be used to form the ‘hands’ of the SCADA system and assist in automating the process.

However, before any automation or remote monitoring can be achieved, the information that is passed to and from the field data interface devices must be converted to a form that is compatible with the language of the SCADA system. To achieve this, some form of electronic field data interface is required. RTUs (Remote Telemetry Units), provide this interface. They are primarily used to convert electronic signals received from field interface devices into the language (known as the communication protocol) used to transmit the data over a communication channel.

The instructions for the automation of field data interface devices are usually stored locally. This is mainly due to the limited bandwidth typical of communications links between the SCADA central host computer and the field data interface devices.

3.3.1.2 Communication Network

The communication network is intended to provide the means by which data can be transferred between the central host computer servers and the field-based RTUs. The Communication Network refers to the equipment needed to transfer data to and from different sites. The medium used can either be cable, telephone or radio [92].

Historically, SCADA networks have been dedicated networks; however, with the increased deployment of office LANs and WANs as a solution for interoffice computer networking, there exists the possibility to integrate SCADA LANs into everyday office computer networks.

The principal advantage of this arrangement is that there is no need to invest in a separate computer network for SCADA operator terminals. In addition, there is an

easy path to integrating SCADA data with existing office applications, such as spreadsheets, work management systems, data history databases, Geographic Information System (GIS) systems, and water distribution modelling systems.

3.3.1.3 Central Host Computer

The central host computer or master station is the one in which the main SCADA software application will be installed, and is most often a single computer or a network of computer servers that provides a man-machine operator interface to the whole SCADA system. The computers process the information received from and sent to the RTU sites and present it to human operators in a form that the operators can work with. Operator terminals are connected to the central host computer by a LAN/WAN so that the viewing screens and associated data can be displayed for the operators.

Recent SCADA systems are able to offer high resolution computer graphics to display a graphical user interface or mimic the screen of the setup. However, with the increased use of the personal computer, computer networking has become commonplace in the office, and, as a result, SCADA systems are now available that can network with office-based personal computers. Indeed, many of today's SCADA systems can reside on computer servers that are identical to those servers and computers used for traditional office applications. This has opened a range of possibilities for the linking of SCADA systems to office-based applications such as monitoring systems, GIS systems, hydraulic modelling software, drawing management systems, work scheduling systems, and information databases [93].

3.3.1.4 Operator Workstations and Software Components

Operator workstations are most often computer terminals that are networked with the SCADA central host computer. The central host computer acts as a server for the SCADA application, and the operator terminals are clients that request and send information to the central host computer based on the request and action of the operators.

An important aspect of every SCADA system is the computer software used within the system. The most obvious software component is the operator interface or Man

Machine Interface/Human Machine Interface (MMI/HMI) package; however, software of some form pervades all levels of a SCADA system. Depending on the size and nature of the SCADA application, software can be a significant cost item when developing, maintaining, and expanding a SCADA system. When software is well defined, designed, written, checked, and tested, a successful SCADA system will likely be produced. Poor performances in any of these project phases will very easily cause a SCADA project to fail [94].

3.3.1.5 Software products typically used within a SCADA system

- **Central host computer operating system:** Software used to control the central host computer hardware. The software can be based on UNIX or any other popular operating system.
- **Operator terminal operating system:** Software used to control the central host computer hardware. The software is usually the same as the central host computer operating system. This software, along with that for the central host computer, usually contributes to the networking of the central host and the operator terminals.
- **Central host computer application:** Software that handles the transmittal and reception of data to and from the RTUs and the central host. The software also provides the graphical user interface which offers site mimic screens, alarm pages, trend pages, and control functions.
- **Operator terminal application:** Application that enables users to access information available on the central host computer application. It is usually a subset of the software used on the central host computers.
- **Communications protocol drivers:** Software that is usually based within the central host and the RTUs, which is required to control the translation and interpretation of the data between ends of the communications links in the system. The protocol drivers prepare the data for use either at the field devices or the central host end of the system.
- **Communications network management software:** Software required to control the communications network and to allow the communications networks themselves to be monitored for performance and failures.

- **RTU automation software:** Software that allows engineer to configure and maintain the application housed within the RTUs (or PLCs). Most often this includes the local automation application, and any data processing tasks that are performed within the RTU.

The above mentioned software products provide the building blocks for the application-specific software, which must be defined, designed, written, tested, and deployed for each SCADA system.

3.3.2 SCADA Architecture

SCADA systems have evolved in parallel with the growth and sophistication of modern computing technology. The following sections will provide a description of the following three generations of SCADA systems:

- First Generation – Monolithic
- Second Generation – Distributed
- Third Generation – Networked

3.3.2.1 Monolithic SCADA Systems

When SCADA systems were first developed, the concept of computing in general centred on 'mainframe' systems. Networks were generally non-existent, and each centralized system stood alone. As a result, SCADA systems were standalone systems with virtually no connectivity to other systems.

The Wide Area Networks (WANs) that were implemented to communicate with remote terminal units (RTUs) were designed with a single purpose in mind – that of communicating with RTUs in the field, and nothing else. In addition, WAN protocols in use today were largely unknown at that time [95].

Redundancy in these first generation systems was accomplished by the use of two identically equipped mainframe systems, a primary and a backup, connected at the bus level. The standby system's primary function was to monitor the primary and take over in the event of a detected failure. This type of standby operation meant that little

or no processing was done on the standby system. Figure 3.3.2-1 shows typical first generation SCADA architecture.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.3.2-1. First Generation SCADA Architecture [95].

3.3.2.2 Distributed SCADA Systems

The next generation of SCADA systems took advantage of developments and improvement in system miniaturization and Local Area Networking (LAN) technology to distribute the processing across multiple systems. Multiple stations, each with a specific function, were connected to a LAN and shared information with each other in real-time. These stations were typically of the mini-computer class, smaller and less expensive than their first generation processors.

Some of these distributed stations served as communications processors, primarily communicating with field devices such as RTUs. Some served as operator interfaces, providing the human-machine interface (HMI) for system operators. Still others served as calculation processors or database servers. The distribution of individual SCADA system functions across multiple systems provided more processing power for the system as a whole than would have been available in a single processor. The networks that connected these individual systems were generally based on LAN protocols and were not capable of reaching beyond the limits of the local environment. Figure 3.3.2-2 depicts typical second generation SCADA architecture.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.3.2-2. Second Generation SCADA Architecture [95].

Distribution of system functionality across network-connected systems served not only to increase processing power, but also to improve the redundancy and reliability of the system as a whole. Rather than the simple primary/standby failover scheme that

was utilized in many first generation systems, the distributed architecture often kept all stations on the LAN in an online state all of the time.

3.3.2.3 Networked SCADA Systems

The current generation of SCADA master station architecture is closely related to that of the second generation, with the primary difference being that of open-system architecture rather than a vendor-controlled, proprietary environment. There are still multiple networked systems, sharing master station functions. There are still RTUs utilizing protocols that are vendor-proprietary. The major improvement in the third generation is that of opening the system architecture, utilizing open standards and protocols, and making it possible to distribute SCADA functionality across a WAN and not just a LAN.

Open standards eliminate a number of the limitations of previous generations of SCADA systems. The utilization of off-the-shelf systems makes it easier for the user to connect third party peripheral devices to the system and/or the network.

The major improvement in third generation SCADA systems comes from the use of WAN protocols, such as the Internet Protocol (IP), for communication between the master station and communications equipment. This allows the portion of the master station that is responsible for communications with the field devices to be separated from the master station 'proper' across a WAN. Vendors are now producing RTUs that can communicate with the master station using an Ethernet connection. Figure 3.3.2-3 represents a networked SCADA system.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.3.2-3. Third Generation SCADA System [95].

Another advantage brought about by the distribution of SCADA functionality over a WAN is that of disaster survivability. The distribution of SCADA processing across a LAN in second-generation systems improves reliability, but in the event of a total loss of the facility housing the SCADA master, the entire system could be lost as well. By

distributing the processing across physically separate locations, it becomes possible to build a SCADA system that can survive a total loss of any one location.

3.3.3 SCADA Protocols

This text has been removed by the author of this thesis for copyright reasons.

3.3.3.1 IEC 60870-5-101

This text has been removed by the author of this thesis for copyright reasons.

3.3.3.2 Distributed Network Protocol version 3 (DNP3)

Protocols define the rules by which devices talk with each other, and DNP3 is a protocol for transmission of data from point A to point B using serial communications. It has been used primarily by utilities like the electricity companies, but it operates suitably in other areas.

The DNP3 is specifically developed for inter-device communication involving SCADA RTUs, and provides for both RTU-to-IED and master-to-RTU/IED. It is based on the three-layer enhanced performance architecture (EPA) model contained in the IEC 60870-5 standards, with some alterations to meet additional requirements of a variety of users in the electric utility industry.

DNP3 is a protocol that fits well into the data acquisition world. It transports data as generic values, has a rich set of functions, and was designed to work in a wide area communications network. The standardized approach and public availability make DNP3 a protocol to be the standard for SCADA applications [97].

3.3.4 Security of SCADA System

SCADA systems have evolved in recent years and are now based on open standards. Most SCADA software and hardware vendors have embraced Transmission Control Protocol/Internet Protocol (TCP/IP) and Ethernet communications, and many have encapsulated their proprietary protocols in TCP/IP packets.

In today's network environment, internal networks are used for communications, including SCADA. SCADA systems are therefore vulnerable to many of the same threats as any TCP/IP-based system. SCADA administrators and industrial systems analysts are often deceived into thinking that since their industrial networks are on

separate systems from the corporate network, they are safe from outside attacks. PLCs and RTUs are usually polled by other third party vendor-specific networks and protocols like RS-232, RS-485, MODBUS, and DNP, and are usually accessed over phone lines, leased private frame relay circuits, satellite systems, licensed and spread spectrum radios, and other token-ring bus topology systems. This often gives the SCADA system administrators a false sense of security since they assume these end devices are protected by these non-corporate network connections [98].

Developing an appropriate SCADA security strategy involves analysis of multiple layers of both the corporate network and SCADA architectures including firewalls, proxy servers, operating systems, application system layers, communications, and policy and procedures. Strategies for SCADA security should complement the security measures implemented to keep the corporate network secure.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.3.4-1. Relationship Between Corporate and SCADA Networks [99].

Figure 3.3.4-1 illustrates the typical network 'ring of defences' and its relationship with the SCADA network. Successful attacks can originate from either internet paths through the corporate network to the SCADA network, or from internal attacks from within the corporate office. Alternatively, attacks can originate from within the SCADA network from either upstream (applications) or downstream (RTUs) paths. What is an appropriate configuration for one installation may not be cost effective for another. Flexibility and the employment of an integrated and coordinated set of layers are critical in the design of a security approach.

3.3.5 Summary

SCADA systems have been used for years in the utilities industry with great success. Today's SCADA systems are able to take advantage of the evolution from mainframe based to client/server architectures. These systems use common communications protocols like Ethernet and TCP/IP to transmit data from the field to the central master control unit.

SCADA protocols have also evolved from closed proprietary systems to an open system, allowing designers to choose equipment that can help them monitor their unique system using equipment from a variety of sources. SCADA systems are widely used to monitor and control critical infrastructure utilities.

Key performance, important features, architectures, protocols and security of SCADA systems have been addressed in the above sections of this chapter. The detailed description of SCADA software application, implementation, and development as an anaesthesia monitoring system for the generation of alerts during anaesthesia are provided in the following sections.

3.4 Supervisory Control and Data Acquisition (SCADA) Software

3.4.1 Introduction

SCADA software can be divided into two types: proprietary or open. Companies develop proprietary software to communicate to their hardware. These systems are sold as 'turnkey' solutions. The main problem with this system is the overwhelming reliance on the supplier of the system. Open software systems have gained popularity because of the interoperability they bring to the system. Interoperability is the ability to mix different manufacturers' equipment on the same system.

Citect and WonderWare are two of the common open software packages available in the market for SCADA systems. Some packages now include asset management integrated within the SCADA system. The details of the SCADA software used in this project work are discussed below.

- Provider – Wonderware
- Name – InTouch
- Version – 9.5
- Operating Systems – Windows 2000 and Windows XP

The Wonderware InTouch is the quickest and easiest way to create human-machine interface (HMI) applications for the Microsoft Windows 2000 and Windows XP operating systems. InTouch is a component of the Wonderware FactorySuite. InTouch applications span the globe in a multitude of vertical markets including food

processing, semiconductors, oil and gas, automotive, chemical, pharmaceutical, pulp and paper, transportation, utilities, and more.

By using InTouch, the programmer can create powerful, full-featured applications that exploit the key features of Microsoft Windows, including activeX controls, OLE, graphics, networking, and more. InTouch can also be extended by adding custom ActiveX controls, wizards, generic objects, and creating InTouch quick script extensions.

3.4.2 Main parts of InTouch software

InTouch consists of three major programs:

- Application Manager
- WindowMaker
- WindowViewer

3.4.2.1 Application Manager

This organises the applications created by the user. It is also used to configure WindowViewer as an NT service, to configure Network Application Development (NAD) for client-based and server-based architectures, and to configure Dynamic Resolution Conversion (DRC) and/or distributed alarming.

3.4.2.2 WindowMaker

This is the development environment where object-oriented graphics are used to create animated, touch-sensitive display windows. These display windows can be connected to industrial I/O systems and other Microsoft Windows applications.

3.4.2.3 WindowViewer

This is the runtime environment used to display the graphic windows created in WindowMaker. WindowViewer executes InTouch QuickScripts, performs historical data logging and reporting, processes alarm logging and reporting, and can function as a client and a server for both DDE and SuiteLink communication protocols.

3.4.3 Features of SCADA software

- User interface
    - Keyboard
    - Mouse
    - Trackball
    - Touch screen
- Graphics displays
    - Customer-configurable, object orientated and bitmapped
    - Unlimited number of pages
    - Resolution: up to 1280 × 1024 with millions of colours
- Alarms
    - Client server architecture
    - Time stamped alarms to 1 millisecond precision (or better)
    - Single network acknowledgment and control of alarms
    - Alarms are shared to all clients
    - Alarms displayed in chronological order
    - Dynamic allocation of alarm pages
    - User-defined formats and colours
    - Up to four adjustable trip points for each analogue alarm
    - Deviation and rate of change monitoring for analogue alarms
    - Selective display of alarms by category (256 categories)
    - Historical alarm and event logging
    - Context-sensitive help
    - On-line alarm disable and threshold modification
    - Event-triggered alarms
    - Alarm-triggered reports
    - Operator comments can be attached to alarms
- Trends
    - Client server architecture
    - True trend printouts, not screen dumps
    - Rubber band trend zooming
    - Export data to DBF, CSV files
    - X/Y plot capability

    - Event based trends
    - Pop-up trend display
    - Trend gridlines or profiles
    - Background trend graphics
    - Real-time multi-pen trending
    - Short- and long-term trend display
    - Length of data storage and frequency of monitoring can be specified on a per-point basis
    - Archiving of historical trend data
    - On-line change of time-base without loss of data
    - On-line retrieval of archived historical trend data
    - Exact value and time can be displayed
    - Trend data can be graphically represented in real-time
- RTU (and PLC) interface
    - All compatible protocols included as standard
    - DDE drivers supported
    - Interface also possible for RTUs, loop controllers, bar code readers, and other equipment
        - Driver toolkit available
        - Operates on a demand basis instead of the conventional predefined scan method
    - Optimization of block data requests to PLCs
    - Rationalization of network user data requests
    - Maximization of PLC highway bandwidth
- Scalability
    - Additional hardware can be added without replacing or modifying existing equipment
    - Limited only by the PLC architecture (typically 300 to 40 000 points)
- Access to data
    - Direct, real-time access to data by any network user
    - Third-party access to real-time data, e.g. Lotus 123 and Excel
    - Network DDE
    - DDE compatibility: read, write and exec

    - DDE to all IO device points
    - Clipboard
- Database
    - ODBC driver support
    - Direct SQL commands or high level reporting
- Networking
    - Supports all NetBIOS compatible networks such as NetWare, LAN Manager, Windows for Workgroups, Windows NT (changed from existing)
    - Support protocols NetBEUI, IPX/SPX, TCP/IP and more
    - Centralized alarm, trend and report processing – data available from anywhere in the network
    - Dual networks for full LAN redundancy
    - No network configuration required (transparent)
    - May be enabled via single check box, no configuration
    - LAN licensing is based on the number of users logged onto the network, not the number of nodes on the network
    - No file server required
    - Multi-user system, full communication between operators
    - RAS and WAN supported with high performance
    - PSTN dial up support
- Fault tolerance and redundancy
    - Dual networks for full LAN redundancy
    - Redundancy can be applied to specific hardware
    - Supports primary and secondary equipment configurations
    - Intelligent redundancy allows secondary equipment to contribute to processing load
    - Automatic changeover and recovery
    - Redundant writes to PLCs with no configuration
    - Mirrored disk I/O devices
    - Mirrored alarm servers
    - Mirrored trend servers
    - File server redundancy

    - No configuration required, may be enabled via single check box, no configuration
- Client/server distributed processing
    - Open architecture design
    - Real-time multitasking
    - Client/server fully supported with no user configuration
    - Distributed project updates (changes reflected across network)
    - Concurrent support of multiple display nodes
    - Access any tag from any node
    - Access any data (trend, alarm, report) from any node

3.4.4 Creating a Monitoring System Application

The InTouch Application Manager is used to create new applications, open existing applications in either WindowMaker or WindowViewer, delete applications, and run the different local utility programs.

3.4.4.1 To create a new application

- On the **File** menu, click **New**, or click the **New** tool in the toolbar. The **Create New Application** wizard appears.
- Click **Next**. A second **Create New Application** wizard appears. By default, the system will display the path to your **InTouch** directory followed by '**NewApp'**.
- In the input box, type the path to the directory in which you want your application to be created or click **Browse** to locate the directory.
- In the **Name** box, type a unique name for the new application's icon that appears when the application is listed in the **InTouch Application Manager** window.

Click **Finish**. The **InTouch - Application Manager** reappears displaying an icon with the name you specified for the new application Figure 3.4.4-1 shows the application manager window.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.4.4-1. The application manager window.

When the new application is created in the application manager of the **InTouch** software, the development of the proposed system is carried out in the window maker programme of this software.

3.4.5 Building a SCADA Monitoring System (SMS) Application

WindowMaker is the development environment for InTouch. The WindowMaker's Application Explorer provides a powerful, graphical method for navigating and configuring the InTouch applications. It provides easy access to WindowMaker's most commonly used commands and functions, such as all Windows commands, all configuration commands, and all InTouch QuickScript editors.

3.4.5.1 Model overview

The aim of this system is to monitor and alert for hypovolaemia in patients during anaesthesia/surgery. Four important physiological parameters are selected, namely, HR, BP, PV and $ETCO_2$. Table 3.4.5-1 shows general conditions for absolute hypovolaemia.

Table 3.4.5-1. General conditions for absolute hypovolaemia.

	Heart Rate (HR)	Blood Pressure (BP)	Pulse Volume (PV)	End Tidal CO2 (ETCO2)
Absolute hypovolaemia	↑	↓	↓	~

Where,

↑ indicates increase in the physiological parameter

↓ indicates decrease in the physiological parameter

~ indicates a variable physiological parameter.

The development was carried out in a stage-by-stage approach for a better and more robust working of the system.

3.4.6 Tagnames

A very important step in the development is assigning the tagname and tag ID for each variable. Tagnames must be defined and assigned a specific type for each one, according to its usage. For example, if the tagname is to read and/or write values coming to or from another physiological feature such as an input BP, it must be an I/O type tagname. The following describes each tagname type and its usage. Figure 3.4.6-1 shows the tagname dictionary in which all assigned tagnames will be stored.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.4.6-1. Tagname dictionary window.

3.4.6.1 Memory Type Tagnames

This text has been removed by the author of this thesis for copyright reasons.

3.4.6.2 Input/Output Type Tagnames

This text has been removed by the author of this thesis for copyright reasons.

Figure 3.4.6-2 shows all types of tags.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.4.6-2. Different tag types available in SCADA software and the marked tag (I/O Integer) were used in the SMS.

3.4.7 Graphical User Interface (GUI) panel

The main aims while designing the GUI display panel for the system are:

- It should be a simple panel
- Contains all the necessary information
- Easy to understand
- Easy to use

Some of the inbuilt tools used for the GUI development are discussed below.

3.4.7.1 Alarms

There are a number of terms and concepts that apply to alarms generally, regardless of the particular system in which they are used, or how they are implemented. In this system this is one of the most important features on which the accuracy of the system depends. In general, an **Alarm** is a specific type of the more general concept of a **condition** — in particular; an alarm is an ***abnormal condition***. Usually, the intention of an alarm is to signal that something has gone wrong, or that a particular stage of processing has been reached. In the first stage the alarm is set on each physiological parameter separately after verifying the limits for the offline analysis of all the data.

3.4.7.2 Alarm Priority

A level of **severity** or **priority** is associated with an alarm to indicate how 'bad' the situation is or, how 'important' the condition is. The priorities of the main four physiological parameters are set first as the most important, and afterwards, all others.

Table 3.4.7-1. Alarm severity and priority range.

This image has been removed by the author of this thesis for copyright reasons.

3.4.7.3 Alarm Types

InTouch software classifies alarms into several general categories based on their characteristics. These categories are known as *Class* and *Type*. The Distributed Alarm System categorizes all alarms into five general *Conditions*: Discrete, Deviation, Rate-of-Change, Value, and SPC. The table below summarizes the alarm condition for both systems:

Table 3.4.7-2. Description of different type of alarms used in the SMS.

This image has been removed by the author of this thesis for copyright reasons.

The basic version of the GUI contains the real time alarm chart and historical trend graph as shown in the figure below.

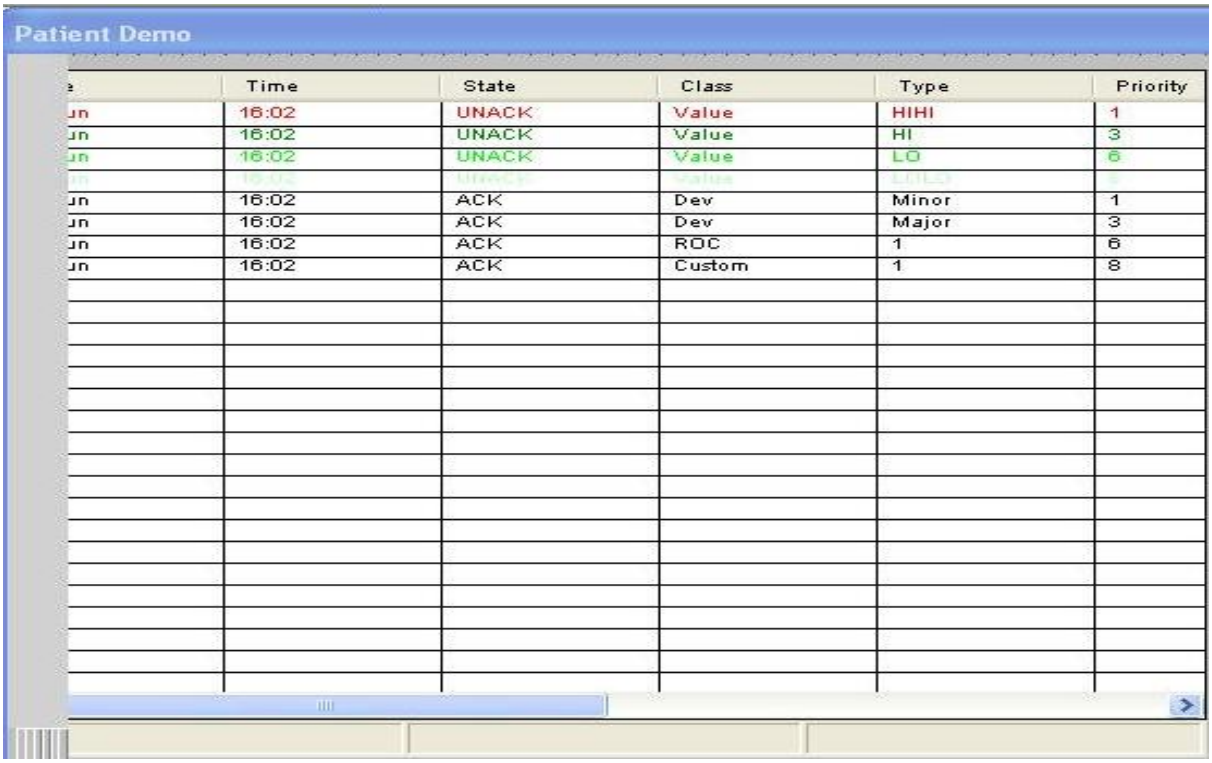


Figure 3.4.7-1. Alarms chart of SMS.

3.4.7.4 Historical Graphical Trend

Historical trends provide a 'snapshot' of data from a time and date in the past. They are not dynamic. Unlike real-time trends, historical trends are only updated when they are instructed to do so, either through the execution of a QuickScript or an action by the operator. Four tagnames (pens) are created for the real time graphical trend.

Figure 3.4.7-2 shows the historical graphical trend tool taken in the GUI of the system.

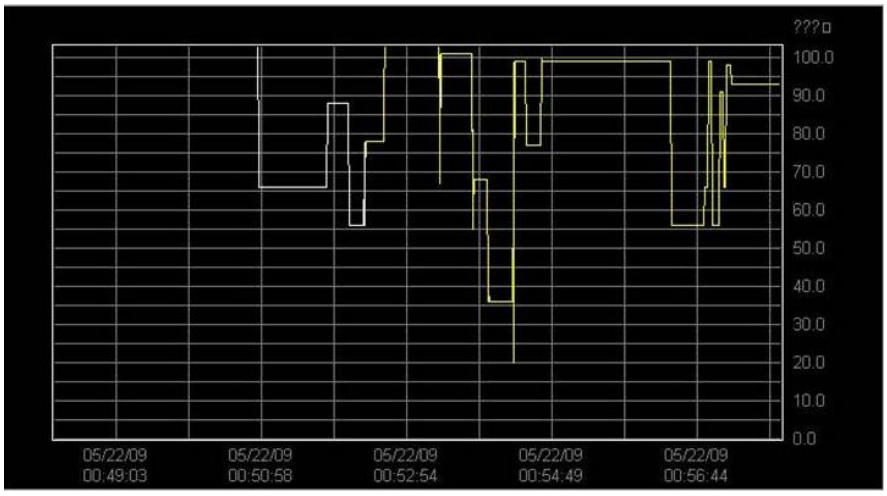


Figure 3.4.7-2. Historical graphical trend chart used in the GUI of SMS.

3.4.7.5 Display Lights

For each physiological parameter one light bulb type of tool is used with its real value shown. When the value exceeds the threshold value of each feature, the light will start to flash; this indicates that the limit of the particular feature exceeds the threshold value.Figure 3.4.7-3 shows the alarm display lights in its off state and Figure 3.4.7-4 shows the alarm display lights in its on state.

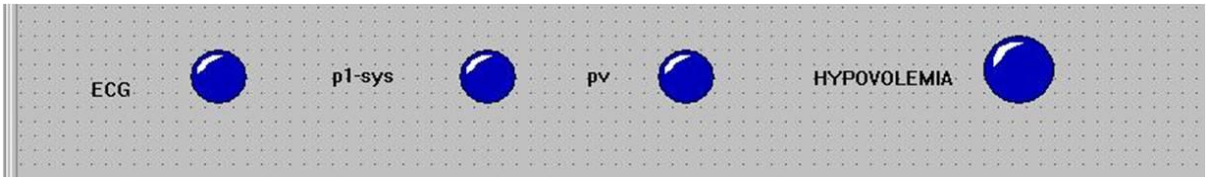


Figure 3.4.7-3. Display lights in off state.

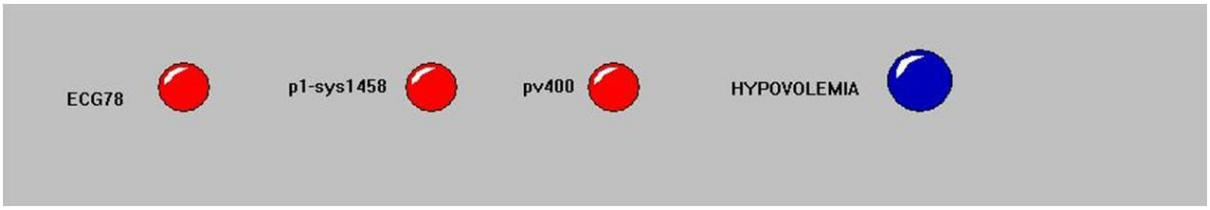


Figure 3.4.7-4. Display lights in on state.

3.4.7.6 Addition in the GUI

Figure 3.4.7-5 is the display of the handy tool for monitoring any one important feature separately from all others. This power tool shows the real-time value, alarm display, and percentage change. This power tool is added to the GUI of the system.

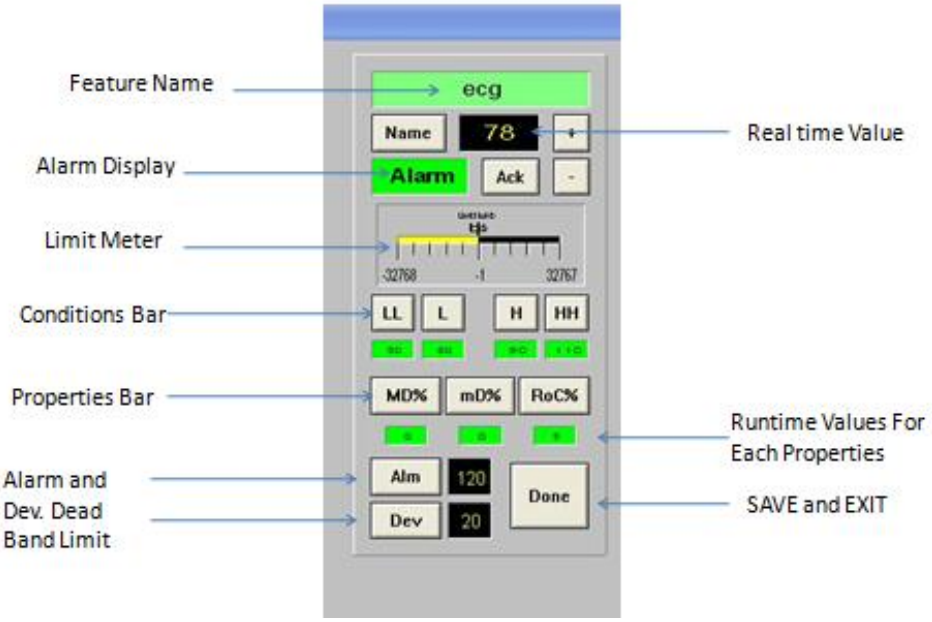


Figure 3.4.7-5. Power tool of the display panel of SMS.

3.4.8 Creating Animation Links

Once the graphic objects or symbols are created, all the graphics can 'bring it to life' by animating it. By attaching animation links to objects or symbols, it changes in appearance to reflect changes in the value of a tagname or an expression.

InTouch supports two basic types of links: Touch Links and Display Links. Touch Links allow operator input into the system. Display Links allow output to the operator. Value sliders or push buttons are examples of Touch Links. Colour fill, location, or blink links are examples of Display Links.

Each feature is linked with all others through different conditions by its tagname, tag id, and unique script. As a result of these links, the simultaneous process is achieved in the runtime state. Figure 3.4.8-1 shows the animation link properties window.

This image has been removed by the author of this thesis for copyright reasons.

Figure 3.4.8-1. Animation links options window in the SMS.

3.4.9 Configuring the Monitoring System

The testing condition for the hypovolaemia is classified as: Mild, Moderate, and Severe. Three different conditions are set before the generation of any alarm in the system. The three conditions are:

3.4.9.1 Condition-1 Threshold Limit

Each physiological parameter is set with its own threshold limit. In the first stage, the threshold limit should exceed all the criteria for this stage of the system before testing for the second condition. Figure 3.4.9-1 shows three parameters are exceeding their threshold limit.

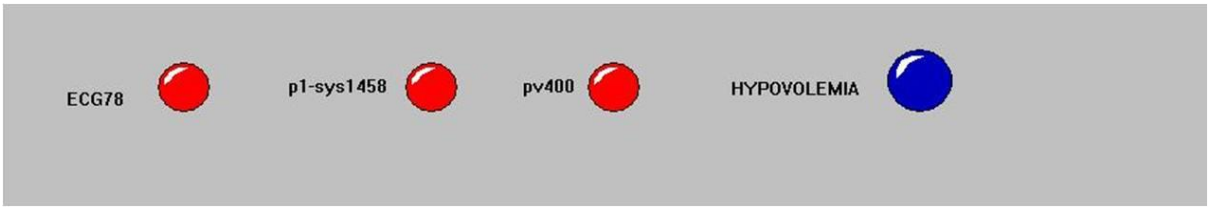


Figure 3.4.9-1. Display lights on the GUI panel of SMS when condition-1 is true.

At this point the system checks for the second condition.

3.4.9.2 Condition-2 Standard Deviation

In this condition, each parameter should exceed its SD limit; this is set after applying normalization method to all the offline data. After exceeding the normal threshold limit (condition-1), the SD limit must be exceeded before the system will check for the third and final condition before any generation of alert or warning.

3.4.9.3 Condition-3 (condition-1 + condition-2)

In this condition, the system checks whether each parameter is true in their normal threshold values (condition-1) and each parameter exceeds the SD limit (condition-2). If both the conditions are true, and then the condition-3 is true, the system gives the alarm/warning in the suitable range of hypovolaemia: mild, moderate, or severe.

Figure 3.4.9-2 shows that when condition-3 is true for all the parameters the alarm for Hypovolaemia is generated.

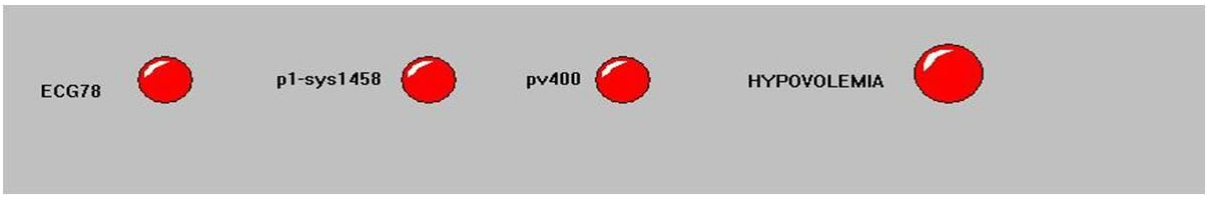


Figure 3.4.9-2. Display lights on the GUI panel of SMS when condition-3 is true.

3.5 Summary

SCADA monitoring system has been developed and tested successfully. Before the final development phase each feature, its tools, script, connectivity, and alarm sensitivity was checked several times with different limits, values, and conditions. The results and testing conditions are discussed in the last chapter of the thesis.

CHAPTER 4 Fuzzy Logic and it's Application to Hypovolaemia Detection

4.1 Introduction

4.1.1 Fuzzy Logic - Background

Fuzzy logic was first proposed by Lotfi A. Zadeh of the University of California at Berkeley in the year 1965 [100]. He elaborated on his ideas in the year 1973 [101] introducing the concept of 'linguistic variables', which he equates to a variable defined as a 'fuzzy set'. The primary objective of fuzzy logic is to map an input space to an output space. The way of controlling this mapping is to use IF-THEN statements known as rules. The order in which these rules are carried out is insignificant, since all rules run concurrently. Other research followed [102, 103], with the first industrial application – a cement kiln built in Denmark – coming on line in 1975. Fuzzy logic is a powerful problem-solving methodology with a myriad of applications in embedded control and information processing. It provides a remarkably simple way to draw definite conclusions from vague, ambiguous, or imprecise information. In a sense, it resembles human decision making with its ability to work with approximate data yet find precise solutions [104].

Unlike classical logic which requires a deep understanding of a system, exact equations, and precise numeric values, fuzzy logic incorporates an alternative way of thinking, which allows modelling complex systems using a higher level of abstraction originating from our knowledge and experience. It allows expressing this knowledge with subjective concepts such as very hot, bright red, and a long time, which are mapped into exact numeric ranges. Fuzzy logic has been gaining increasing acceptance during the past few years. There are over two thousand commercially available products using this logic, ranging from washing machines to high-speed trains. Nearly every application can potentially realize some of the benefits of fuzzy logic, such as performance, simplicity, lower cost, and increased productivity.

4.1.2 Fuzzy Logic – An Example

Fuzzy Logic is a logic that arrives at a definite conclusion based on vague, ambiguous, or imprecise input information. When applying mathematical concepts to our daily lives it is often difficult to adhere to the logical constraints of traditional set theory because of the vagueness of the real world. The logic enables an object to belong to a set with a certain degree; unlike traditional logic, it addresses the complexity of the world. It can also be used practically to aid systems in decision making. To provide a clear understanding of fuzzy logic, we refer to two examples. Let us firstly consider the degree of truth .The statement, 'Today is sunny' may have different degrees of truth. It may be 100 percent true providing there are no clouds, 80 percent true providing there are a few clouds, 50 percent true providing it is hazy and 0 percent true providing it rains all day. Now, let us observe how it aids in decision making. Consider the Statement 'If today is not too hot and not rainy then I will go out to play'. This suitable condition of the statement can be reached by using basic fuzzy propositional logic. It analyses the degree of the suitable condition based on the sets designated hot and rainy, obtains a crisp value, and then finally outputs a definite result. We will indulge more on decision-making in how fuzzy logic works [105].

4.1.3 Fuzzy Logic – Applications

What Zadeh proposed has been more influential than its initial acceptance in the Far East. Its success in many control applications has ensured its adoption around the world. Fuzzy logic has been applied in many areas, gaining wide acceptance in Japan and many other countries including the USA. Household appliances such as dishwashers and washing machines use fuzzy logic to determine the optimal amount of soap, the correct water pressure for dishes and clothes, and it is even used in self-focusing cameras [106].

Fuzzy logic has been successfully applied in many decision and control applications. Some more publications by Zadeh [107, 108] were studied for a more thorough understanding of fuzzy logic and its applications in this study. In another publication, Zadeh [109] demonstrated how robust fuzzy algorithms could be created by exploiting the ability of fuzzy sets to handle imprecise linguistic rules. Zadeh [110] also discussed the use of fuzzy logic and its impact in the domain of soft computing. Soft computing is real life application of computer-based methodologies aimed at accommodating the imprecision in real world computer applications.

4.1.4 Usage of Fuzzy Logic

Today, computers have a brilliant capacity for decision making for crisp processes. However this is limited to systems which have a mathematical interpretation with the lack of human reasoning. Computers use binary logic and, prior to Zadeh, could only allow for values 1 for true and 0 for false. Statements like ‘this car is not fast enough’ or ‘this person is quite smart’ are rather vague statements which cannot be interpreted by classical logic. To handle this vagueness, fuzzy logic provides an extension from the classical logic [111]. Fuzzy logic starts, and builds on, a set of user-supplied human language rules. The fuzzy systems convert these rules to their mathematical equivalents. This simplifies the job of the system designer and the computer, and the results are a much more accurate representation of the way systems behave in the real world.

A fuzzy system can be created to match any set of input-output combinations. The rule inference system of the fuzzy model consists of a number of conditional IF-THEN rules. For the designer who understands the system, these rules are easy to write, and as many rules as are necessary can be supplied to describe the system adequately. The concept of Adaptive Neuro Fuzzy Interference system (ANFIS) [112] as system identification has been used in this project. The fuzzy-logic defuzzification used by ANFIS is based on a zero-order Sugeno fuzzy model (or FIS, Fuzzy Inference System) [113]. The following sections will present and develop ideas such as sets, membership functions, logical operators, linguistic variables, and rule bases.

4.2 Fuzzy Sets, Membership Functions and Logical Operators

Introduction to Fuzzy Sets, Fuzzy Logic, and Logical Operators of the fuzzy control systems establishes a strong foundation for designing and analyzing fuzzy control systems under uncertain and irregular conditions. Mastering its contents gives a clear understanding of fuzzy control systems’ theory that prepares for deeper and broader knowledge, and for many practical challenges faced in modern industry.

4.2.1 Fuzzy Sets

Fuzzy sets are sets without clear or crisp boundaries. The elements they contain may only have a partial degree of membership. They are, therefore, not the same as classical

sets in the sense that the sets are not closed. Some examples of vague fuzzy sets and their respective units include the following.

- Loud noises (sound intensity)
- Ambience (brightness)
- High speeds (velocity)
- Desirable actions (decision of control space)

Fuzzy sets can be combined through fuzzy rules to represent specific actions/behaviour and it is this property of fuzzy logic that will be utilised when implementing a fuzzy logic controller in subsequent sections.

4.2.2 Membership Functions

A membership function (MF) is a curve that defines how each point in the input space is mapped to the set of all real numbers from 0 to 1. This is really the only stringent condition brought to bear on an MF. A classical set may be, for example, written as:

$$A = \{x \mid x > 3\} \quad \textbf{4.1}$$

Now if X is the universe of discourse with elements x then a fuzzy set A in X is defined as a set of ordered pairs:

$$A = \{x, \mu A(x) \mid x \varepsilon X\} \quad \textbf{4.2}$$

Note that in the above expression $\mu A(x)$ may be called the membership function of x in A and that each element of X is mapped to a membership value between 0 and 1. Typical membership function shapes include triangular, trapezoidal and Gaussian functions. The shape is chosen on the basis of how well it describes the set it represents.

Figure 4.2.2-1 shows the example of fuzzy sets created in the triangular shape. In this example the MF's are created as S is small; MS is medium small; M is medium, ML is medium large and L is large. The values of these sets vary from 0 to 1 in both the axes.

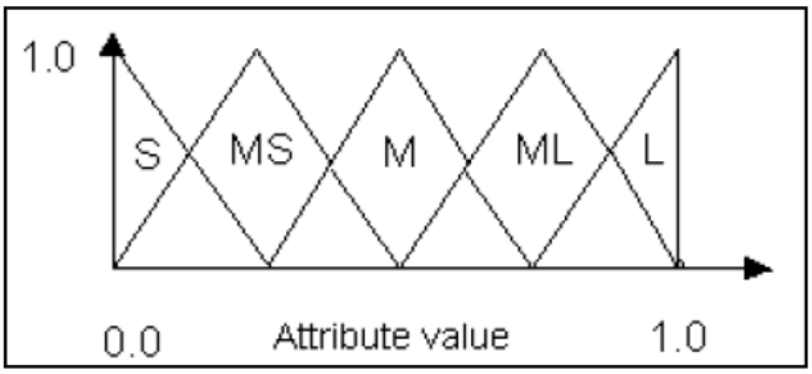


Figure 4.2.2-1. Example Fuzzy Set. S is small; MS is medium small; M is medium, ML is medium large; L is large.

Figure 4.2.2-2 shows the example of fuzzy sets created in Gaussian shape. In this example the MF's are created as poor, good and excellent. The value of these sets on x-axis is 0 to 100 and on y-axis is 0 to 1.

This image has been removed by the author of this thesis for copyright reasons.

Figure 4.2.2-2 Example of a three-part Gaussian shaped MF.

4.2.3 Logical Operators

Fuzzy logic reasoning is a superset of standard Boolean logic, yet it still needs to use logical operators such as AND, OR and NOT. Firstly, note that fuzzy logic differs from Boolean yes/no logic in that although TRUE is given a numerical value '1' and a FALSE numerical value is given '0', other intermediate values are also allowed. For example the values 0.2 and 0.8 can represent both not-quite-false and not-quite-true, respectively.

It will be necessary to do logical operations on these values that lie in the [0, 1] set, but two-valued logic operations like AND, OR and NOT are incapable of doing this. For this functionality, the functions min, max, and additive complement (1-A) will have to be used. Table 4.2.3-1 shows how it works for the OR and NOT logical operations.

Table 4.2.3-1. Truth table (left) for OR operator and truth table (right) for NOT operator.

A	B	A OR B	Max (A,B)
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

A	NOT A	1-A
0	1	1
1	0	0

Below figure shows the two valued and multi-valued logical operations using the truth tables for both the operators.

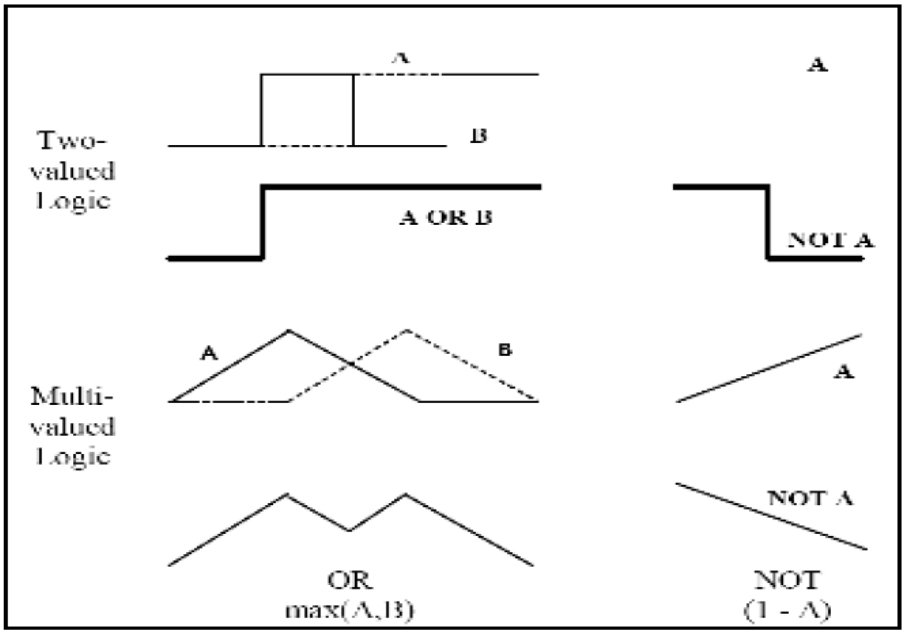


Figure 4.2.3-1. Example of two-valued and multi-valued logical operation.

4.2.4 Linguistic Variable and Rule Bases

Linguistic variables are values defined by fuzzy sets. A linguistic variable such as 'High Speed' for example could consist of numbers that are equal to or between 50km/h and 80km/h. The conditional statements that make up the rules that govern fuzzy logic behaviour use these linguistic variables and have an IF-THEN syntax.

These IF-THEN rules are what make up fuzzy rule bases. A sample IF-THEN rule where A and B represent linguistic variables could be:

If x is A then y is B

The statement is understood to have both a premise, if 'x is A', and a conclusion, then 'y is B'. The premise, also known as the antecedent, returns a single number between 0 and 1 whereas the conclusion, also known as the consequent, assigns the fuzzy set B to the output variable y. Another way of writing this rule using the symbols of assignment '=' and equivalence '==' is:

If x == A then y = B

An IF-THEN rule can contain multiple premises or antecedents. For example,

- IF speed is high and road is wet and brakes are poor THEN….

Similarly, the consequent of a rule may contain multiple parts.

- IF temperature is very high then fan is on and throughput is reduced THEN

Rule bases involve a number of distinct steps such as:

1. Firstly, the inputs must be fuzzified to a degree of membership between 0 and 1. This means that if the antecedent is true to some degree of membership, then the consequent is also true to that same degree.
2. Secondly, fuzzy operators are applied for antecedents with multiple parts to get a single number between 0 and 1.
3. Thirdly, the result is applied to the consequent. This step is also known as implication. The degree of support for the entire rule is used to shape the output fuzzy set.
4. The outputs of fuzzy sets from each rule are aggregated into a single output fuzzy set. This final set is evaluated (or defuzzified) to get a single number.

4.2.5 An Example

The following example in Figure 4.2.5-1 shows how these steps are applied in practice. Consider the two rules for a fuzzy model that evaluates lecture attendance:

1. IF (lecture quality is good) AND (interest of material is interesting) THEN (attendance is high).
2. IF (lecture quality is poor) OR (interest of material is boring) THEN (attendance is low).

Figure 4.2.5-1. Application of rules for lecture attendance example with the output showing attendance in the third column last box with a line on 75.

The fuzzy AND operator is applied in rule one [IF (lecture quality is good) AND (interest of material is interesting) THEN (attendance is high)] and since the premise of the rule is true to a high degree then the consequent is also going to be true to a high degree. In this example both the fuzzy AND operator and the implication operator use the minimum function, hence for an input of 50 percent for Lecture Quality and 80 percent for Interest of Material, the defuzzified attendance percentage works out to be 75 percent .

This comes from;

Min (μLectureQuality, μIntrestOfMaterial)

= min (1.0, 0.75)

= 0.75

Immediate advantages of this approach become apparent. Fuzzy sets can be combined using fuzzy rules to define system behaviour, and thus complex non-linear systems can be expressed linguistically. The process of fuzzifying a single crisp input, applying fuzzy operators and then defuzzifying it to produce a single crisp output is known as fuzzy inference. This progression of modelling is discussed in detail in the next section.

4.3 Fuzzy Logic Models

Standard control techniques use numeric data to relate input and output signals. Fuzzy logic systems can use both numeric information and linguistic information to perform a

mapping of an input to an output space. Consider the following diagram in Figure 4.2.5-1.

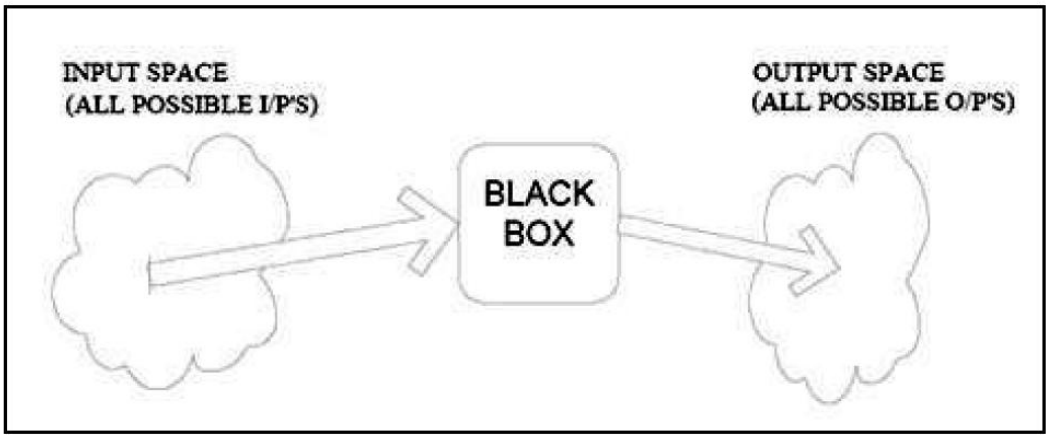


Figure 4.2.5-1. Mapping of an input space to an output space.

Many different control mechanisms could reside within the black box, but in this case the mechanism will be confined to a fuzzy logic system. Since the objective is to map inputs to outputs, it becomes possible to model non-linear systems, even complex ones. This is one of the fuzzy logic's greatest advantages. Put differently, fuzzy logic systems are tolerant of imprecise data. When considered, this suits many real-world applications well because as real-world systems become increasingly complex, often the need for highly precise data decreases. The rules that govern this mapping can be acquired through two methods: the first is a method called the direct approach, and the second is the use of system identification.

The direct approach involves the manual formulation of linguistic rules by a human expert. These rules are then converted into a formal fuzzy system model. The problem with this approach is that unless the human expert knows the system well, it is very difficult to design a fuzzy rule base and inference system that is workable, let alone efficient. For complex systems (non-linear for example) tuning these membership functions would require the adjustment of many parameters simultaneously. Understandably, no human expert could accomplish this.

Fuzzy models that are designed using system identification are based on the use of input/output data. System identification was introduced to overcome the difficulties involved in the direct approach of choosing the fuzzy set's membership functions using a search/optimisation technique to aid the selection. All of the previous elements of

fuzzy logic that have been discussed up to this point are put together to form a fuzzy inference system (FIS). There are two main types of fuzzy inference system:

- The Mamdani type fuzzy system
- The Sugeno type fuzzy system

They are both introduced in the following sections.

4.3.1 Mamdani Modelling

Deriving its name from Ebrahim Mamdani, the Mamdani model was the first efficient fuzzy logic controller designed, and was introduced in 1975 [114]. The controller consists of a fuzzifier, a fuzzy rule base, an inference engine, and a defuzzifier. It is shown in Figure 4.3.1-1.

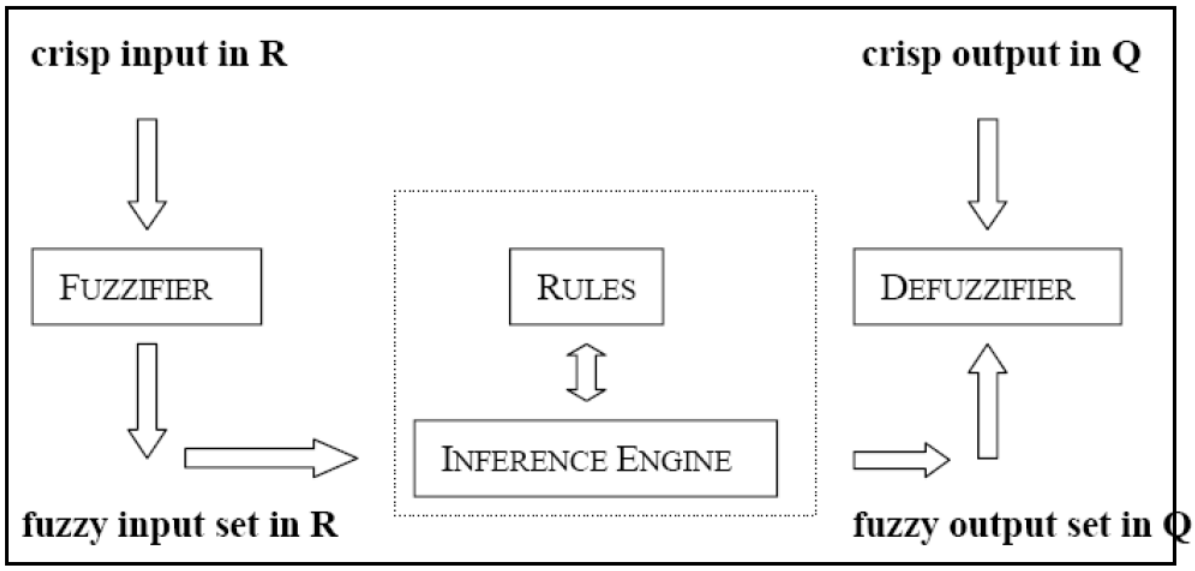


Figure 4.3.1-1. Mamdani Fuzzy Control System.

Conventional control systems require crisp outputs to result from crisp inputs. The above representation shows how a crisp input in R can be operated on by a fuzzy logic system to yield a crisp output in Q. This Mamdani controller [115] is realised using the following steps.

4.3.1.1 Fuzzification of Inputs

The fuzzifier maps crisp input numbers into fuzzy sets. Each input value between 0 and 1 is given, which represents the degree of membership that input has within these

output fuzzy sets. Fuzzification can be implemented using lookup tables or, as is the case in this report, using membership functions.

4.3.1.2 Application of Fuzzy Operators

In the examples where multiple statements are used in the antecedent of a rule, it is necessary to apply the correct fuzzy operators. This allows the antecedent to be resolved to a single number that represents the strength of that rule.

4.3.1.3 Application of Implication Method

This part of the Mamdani system involves defining the consequence as an output fuzzy set. This can only be achieved after each rule has been evaluated and is allowed to contribute its 'weight' in determining the output fuzzy set.

4.3.1.4 Aggregation of all Outputs

The fuzzy outputs of each rule need to be combined in a meaningful way to be of any use. Aggregation is the method used to perform this by combining each output set into a single output fuzzy set. The order of rules in the aggregation operation is unimportant as all rules are considered. The three methods of aggregation available for use include SUM (sum of each rule's output set), MAX (maximum value of each rule's output set) and the probabilistic method (the algebraic sum of each rule's output set). An example of the aggregation process using the max operator can be seen in Figure 4.3.1-2.

4.3.1.5 Defuzzification of Aggregated Output

The aggregated fuzzy set found in the previous step is the input to the defuzzifier. As indicated in the model shown in Figure 4.3.1-2, this aggregated fuzzy set in Q is mapped to a crisp output point in Q. This crisp output is a single number that can be usefully applied in controlling the system. A number of methods of defuzzification are possible, and these include the mean of maximum, largest of maximum, smallest of maximum and centroid (centre of area) methods. The centroid method is the most widely used and can be seen in Figure 4.3.1-2.

Figure 4.3.1-2. Diagram showing aggregation and defuzzification.

4.3.2 Sugeno Modelling

The Sugeno fuzzy model, (or more fully the Takagi-Sugeno-Kang method) of fuzzy inference was first introduced in 1985 [116]. In many respects, it is identical to the Mamdani method except that the output membership functions for the Sugeno method are always linear or constant. The output membership functions can be thought of as singleton spikes that undergo a simple aggregation instead of other aggregation methods such as MAX or SUM. Figure 4.3.2-1 shows the application of three basic rules for a Sugeno model. All three rules have been written using the OR connector.

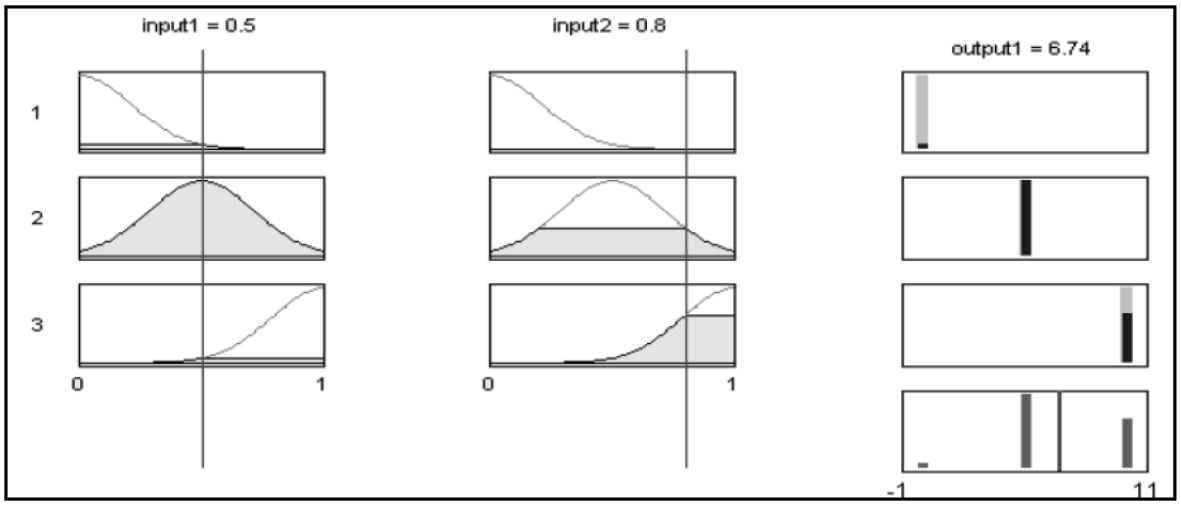


Figure 4.3.2-1. Implementation of Sugeno Model.

For example: if input1 is x or input2 is y then output1 is z

The method of defuzzification is the weighted average (as marked by the thin line in the bottom right corner of the Figure 4.3.2-1. The output is always a single number, in this case inputs are [0.5, 0.8] and one output is [6.64]. The Sugeno system is computationally efficient and its ability to interpolate multiple linear models makes it particularly suited to modelling non-linear systems.

The next section explains different fuzzy logic methods adopted for the generation of alerts in this anaesthesia monitoring project.

4.4 Fuzzy Logic System Development

This section explores the development of a fuzzy logic based monitoring system and the techniques used to make the system more clinically relevant and robust as a real

time system. It includes clustering, FIS modelling, training with ANFIS and Mamdani and Sugeno types of modelling systems.

4.4.1 Clustering

Clustering of numerical data forms the basis of many classification and system modelling algorithms. The purpose of clustering is to identify natural groupings of data from a large data set to produce a concise representation of a system's behaviour.

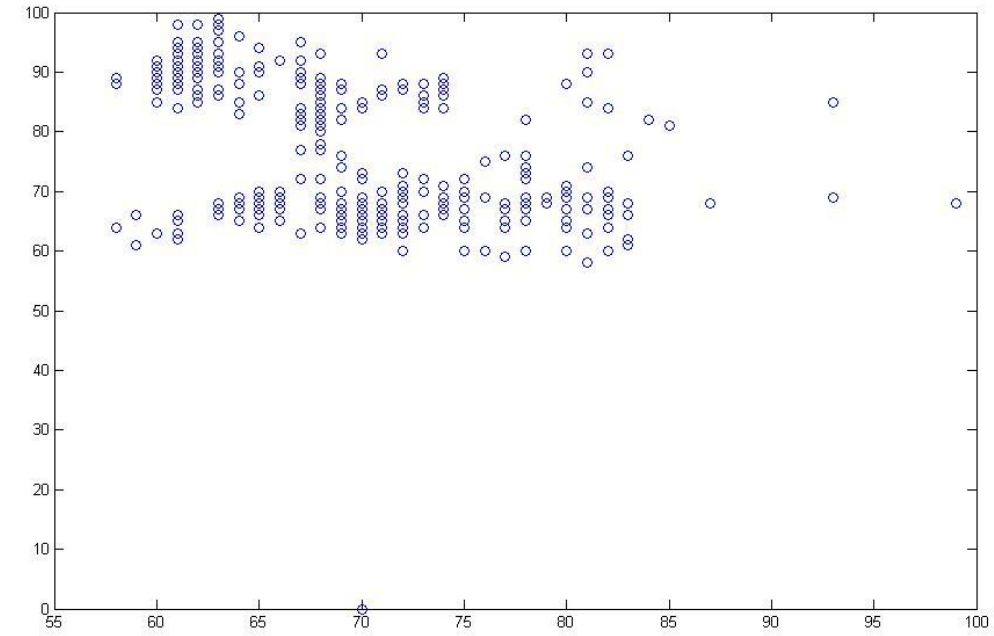
Matlab's Fuzzy Logic Toolbox allows finding the clusters in input-output training data. The cluster information is used to generate a Sugeno-type fuzzy inference system that best models the data behaviour using a minimum number of rules. The rules partition themselves according to the fuzzy qualities associated with each of the data clusters, and the data clustering is of two types:

4.4.1.1 Fuzzy C-Means Clustering

Fuzzy C-Means (FCM) is a data clustering technique wherein each data point belongs to a cluster to some degree that is specified by a membership grade. This technique was originally introduced by Jim Bezdek in 1981 [117] as an improvement on earlier clustering methods. It provides a method that shows how to group data points that populate some multidimensional space into a specific number of different clusters.

Matlab function *fcm* starts with an initial guess for the cluster centres, which are intended to mark the mean location of each cluster. The initial guess for these cluster centres is most likely incorrect. Additionally, FCM assigns every data point a membership grade for each cluster. By iteratively updating the cluster centres and the membership grades for each data point, it iteratively moves the cluster centres to the right location within a data set.

The figures below shows the three stages of FCM using the ECG-heart rate of two different patients; i.e., Patient Number 9 and Patient Number 10. The first stage is the input of data, the second stage is the clustering, and third stage is the output stage where data sets are separated by FCM. Their unique centres and MFs are saved in the digital form to be used in fuzzy modelling.



-

Figure 4.4.1-1. ECG-HR of Patient-9 and y axis ECG-HR of Patient-10.

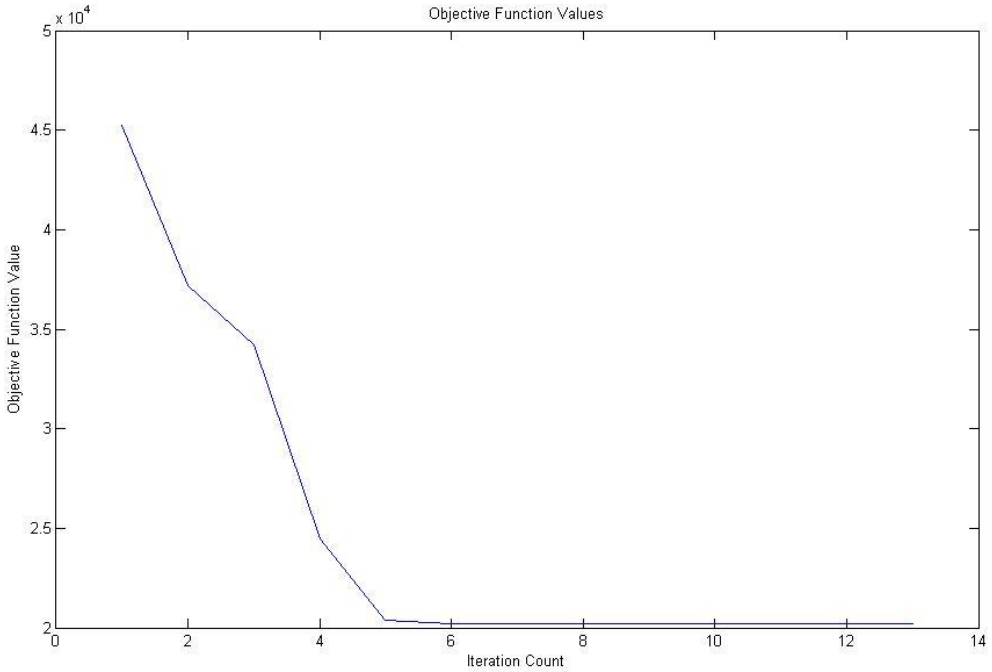


Figure 4.4.1-2. Stage-2(clustering): x axis- number of iteration count and y axis- objective function value.

Figure 4.4.1-3 shows the output from the FCM clustering technique, by iteratively updating the cluster centres and the membership grades for each data point. It moves the cluster centres to the right location within a data set. These centres are the larger characters shown in the cluster centres with circle and cross.

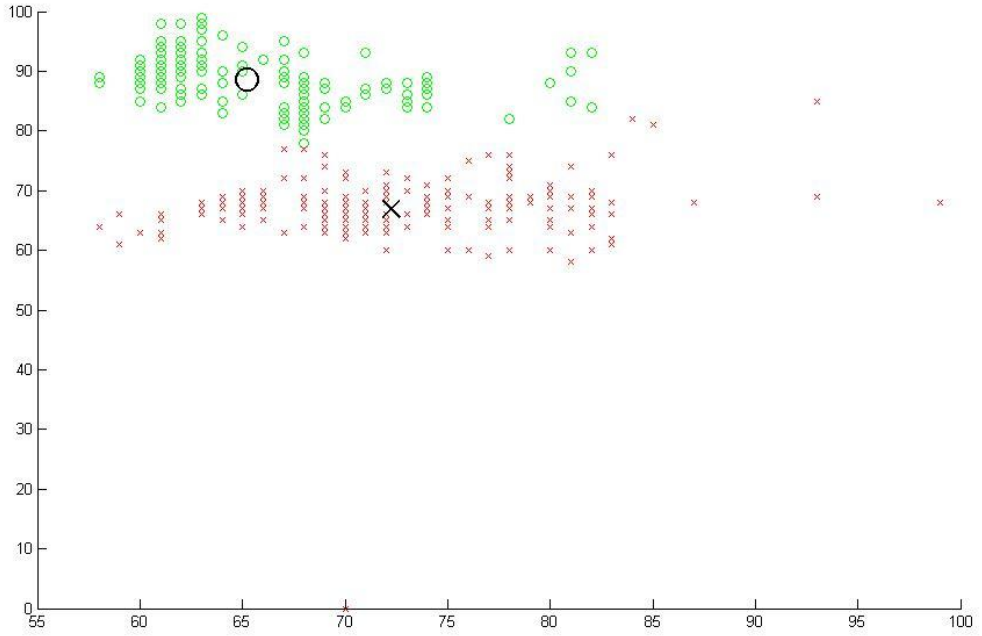


Figure 4.4.1-3. Stage-3(output): x axis- clustered ECG-HR of Patient-9 with the MF's centre as circle in the green circles and y axis- clustered ECG-HR of Patient-10 with the MF's centre as cross in the red crosses.

The FCM outputs a list of cluster centres and several membership grades for each data point. This information, which is returned by FCM, helps in building a fuzzy inference system by creating membership functions to represent the fuzzy qualities of each cluster; this is discussed in detail below.

4.4.1.2 Subtractive Clustering

Subtractive clustering [118] is a fast, one-pass algorithm for estimating the number of clusters and the cluster centres in a set of data. The cluster estimates, which are obtained from the *subclust* function, can be used to initialize iterative optimization-based clustering method like FCM and model identification method like ANFIS. The *subclust* function finds the clusters by using the subtractive clustering method.

This function provides a fast, one-pass method to take input-output training data and generate a Sugeno-type fuzzy inference system that models the data behaviour.

Figure 4.4.1-4 shows the first stage of subtractive clustering, where the data has been loaded. The green wave is the blood pressure of Patient Number 9, the number of the sample on the x-axis and the sample value on the y-axis.

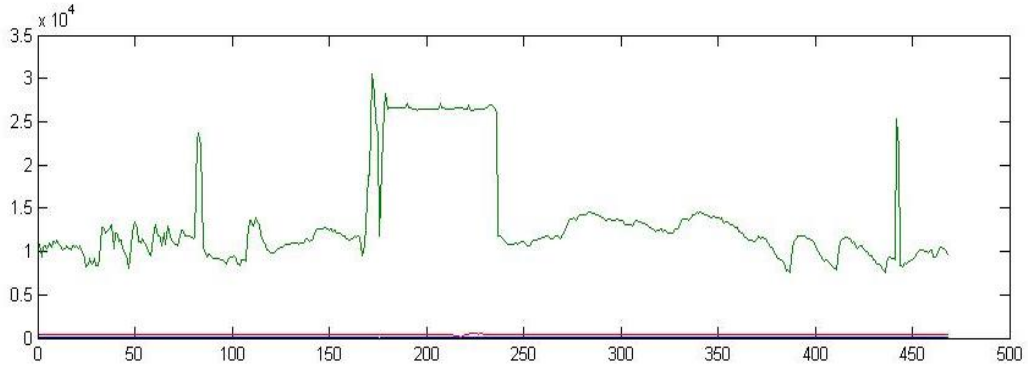


Figure 4.4.1-4. Stage-1 input data loaded with x-axis has number of samples and y-axis has the sample values.

Figure 4.4.1-5 shows the small circles on the loaded data (green coloured) which demonstrates that this can be used as a stand-alone, fast method for generating a fuzzy model from data, or as a pre-processor to ANFIS for determining the initial rules. An important advantage of using this subtractive clustering method is finding and generating rules which are more tailored to the input data than they are in an FIS, generated without clustering. This reduces the problem of an excessive propagation of rules when the input data has a high dimension. This is an important form of data pre-processing; it can then be loaded into the FIS structure for training, testing and checking.

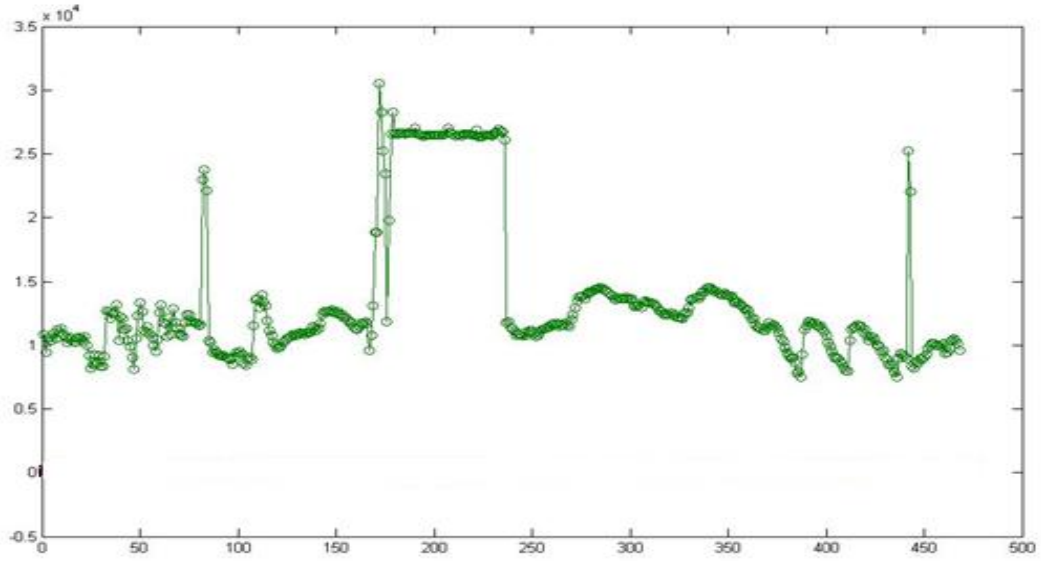


Figure 4.4.1-5. Stage-2 subtractive clustered output wave form where x-axis.

4.5 FIS and ANFIS

4.5.1 Fuzzy Inference Systems (FIS)

Fuzzy inference is the process of formulating the mapping from a given input to an output, using fuzzy logic. The mapping then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference consists of membership functions, logical operations and IF-THEN rules. There are two types of fuzzy inference systems in the Matlab's fuzzy logic toolbox: Mamdani-type and Sugeno-type. These two types of inference systems are explained in previous sections of this chapter.

The Fuzzy inference system is a model that maps input characteristics to input membership functions, input membership function to rules, rules to a set of output characteristics, output characteristics to output membership functions, and the output membership function to a single-value output or a decision associated with the output. Fuzzy inference is the modelling system whose rule structure is essentially predetermined by the user's interpretation of the variables in the model. The Figure 4.5.1-1 shows the general FIS structure.

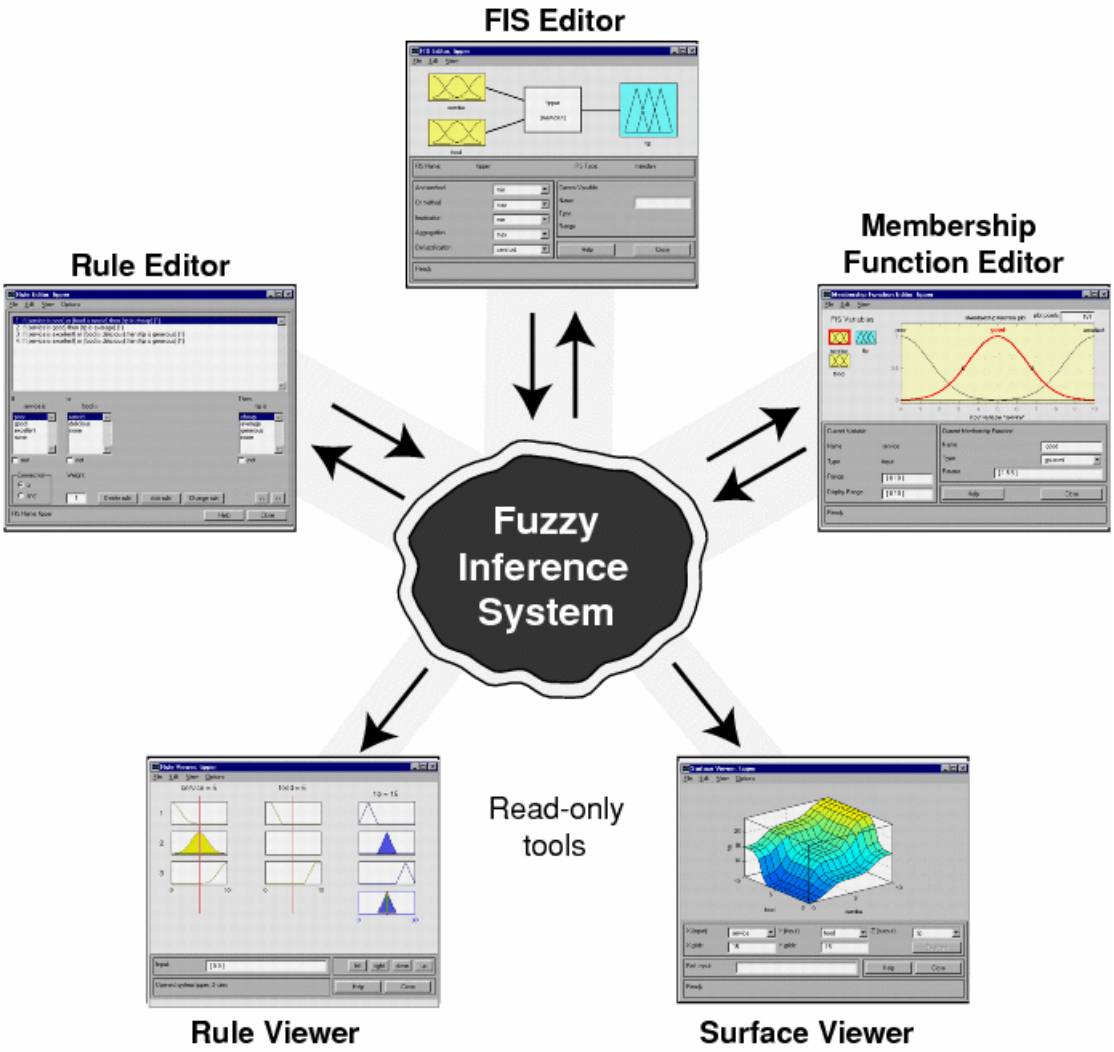


Figure 4.5.1-1. General FIS structure [119].

4.5.2 Adaptive Neuro Fuzzy Inference System (ANFIS)

The acronym ANFIS derives its name from *adaptive neuro-fuzzy inference system*. Using a given input/output data set, the *anfis* function constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a back-propagation algorithm alone or in combination with a least squares type of method. This adjustment allows the fuzzy systems to learn from the input data.

4.5.2.1 Constraints of ANFIS

ANFIS is much more complex than the fuzzy inference systems discussed so far, and is not available for all of the fuzzy inference system options. Specifically, ANFIS only supports Sugeno-type systems, and these must have the following properties:

- Be first or zeroth order Sugeno-type systems.
- Have a single output, obtained using weighted average defuzzification.
- All output membership functions must be of same type and either linear or constant.
- Have no rule sharing. Different rules cannot share the same output membership function, namely the number of output membership functions must be equal to the number of rules.
- Have unity weight for each rule.

An error occurs if your FIS structure does not comply with these constraints.

4.5.3 Training Adaptive Neuro Fuzzy Inference Systems

4.5.3.1 Loading and Plotting the Data

Training an FIS begins with loading the training data set that contains the desired input/output data of the system to be modelled. Any data set loaded must be an array with the data arranged as column vectors, and the output data in the last column. It can also be loaded as testing and checking data in the GUI.

Using the Load Data portion of the GUI, we load the training data set. After loading the training data, a plot is displayed. The training, testing, and checking data are annotated in blue as *circles*, *diamonds*, and *pluses* respectively as shown in Figure 4.5.3-1.

4.5.3.2 Generating the Initial FIS Structure

Specification of an initial FIS model structure is necessary before the start of FIS training.

Figure 4.5.3-1 shows the newly created Sugeno-type FIS structure with three inputs as ECG, BP, and PV and an output as Hypovolaemia.

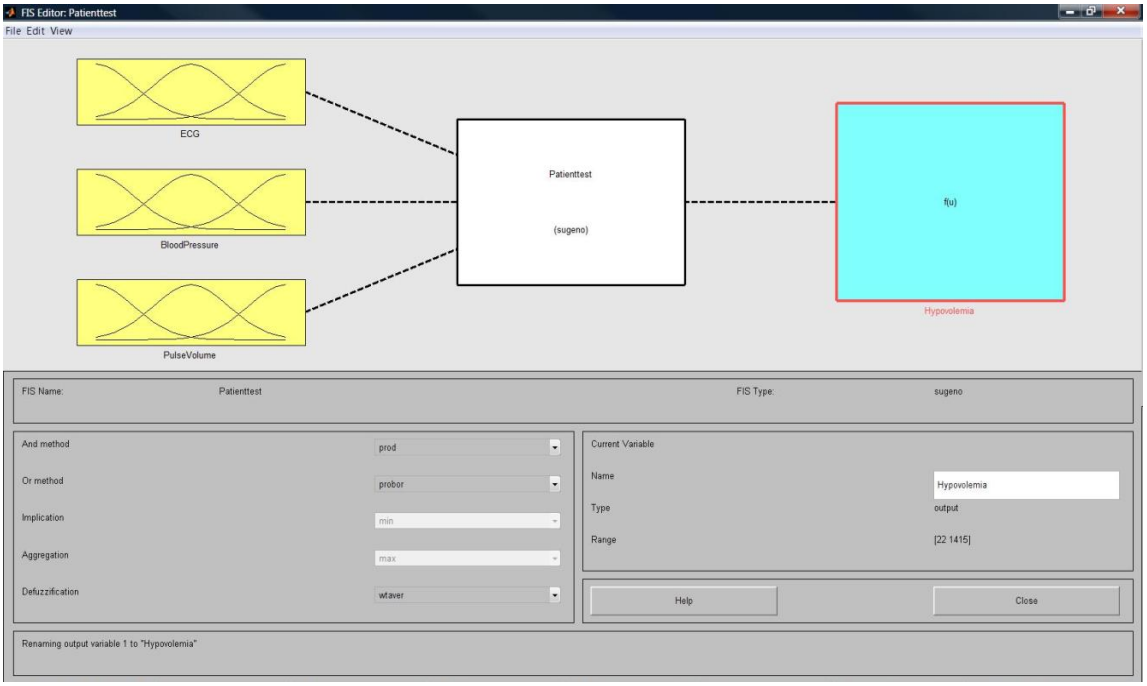


Figure 4.5.3-1. Sugeno-type FIS structure created for generating initial ANFIS structure.

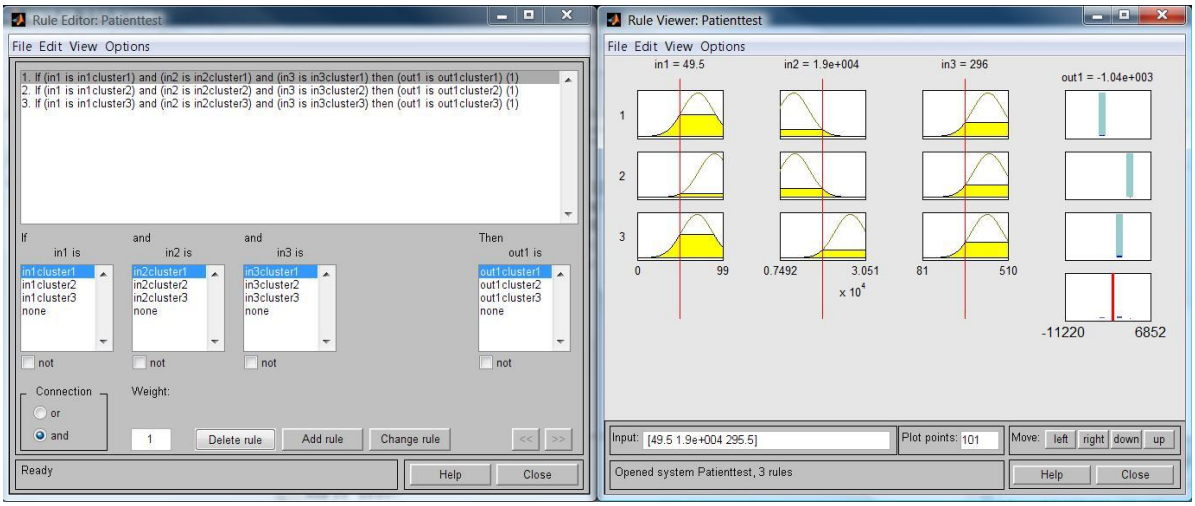


Figure 4.5.3-2 Left: Rule editor window of the above Sugeno-type FIS structure.

To generate the initial FIS model, we used two partitioning techniques:

- Grid partition— it generates a single-output Sugeno-type FIS by using grid partitioning on the data.
- Sub. Clustering — it generates an initial model for ANFIS training by first applying subtractive clustering on the data.

Figure 4.5.3-3 shows the FIS structure generation window; this appears after training data is loaded. It asks the developer to specify the initial input MFs, input MF type and output MF type.

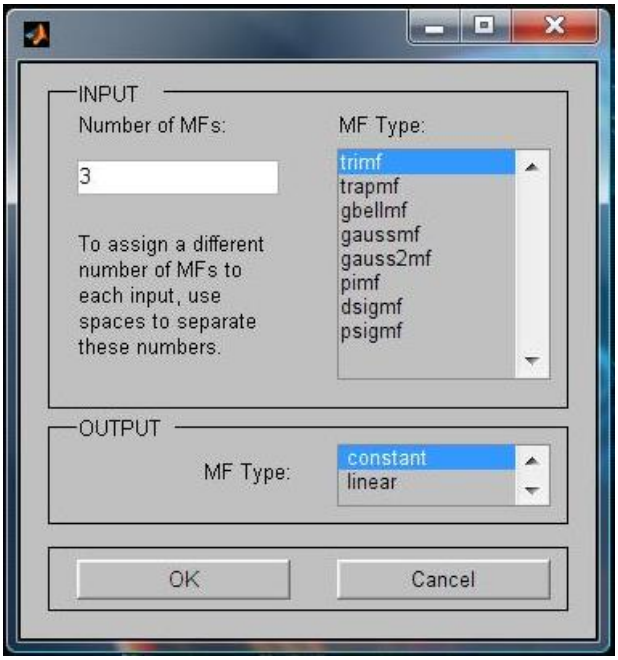


Figure 4.5.3-3. The FIS structure generation window.

Figure 4.5.3-4 shows the ANFIS model structure with rules mapping the output after specifying the initial conditions of the created Sugeno-type FIS system.

From left to right in Figure 4.5.3-4:

- 3 black circles are the given inputs ECG-HR, BP and PV.
- 9 white circles are the self generated input MFs, 3 for each rule, which is 3 rules x 3 MFs.
- 27 blue circles are the self generated rules.
- 27 white circles are the self generated output MFs.
- 1 white circle is the mapping point for all the output MFs.
- 1 black circle is the output hypovolaemia.

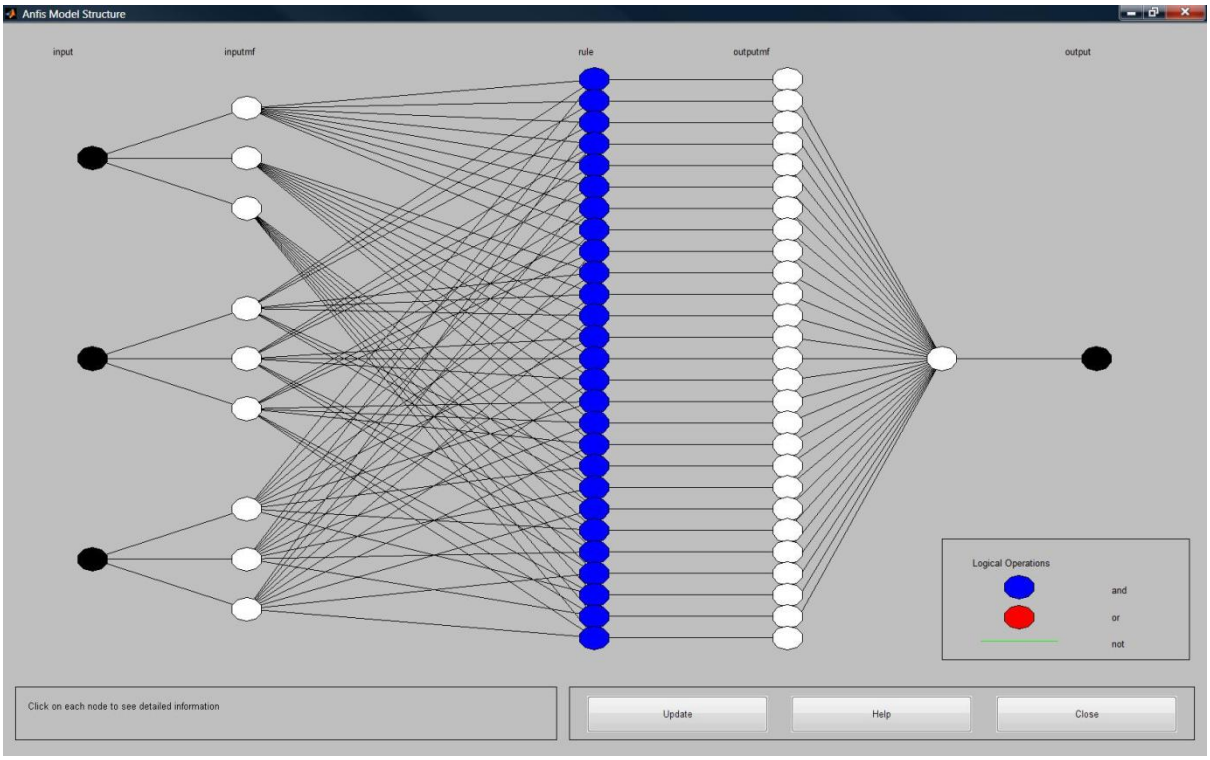


Figure 4.5.3-4. The ANFIS model structure with rules mapping output.

4.5.3.3 Training the FIS

After loading the training data and generating the initial FIS structure, training the FIS can start. The optimization methods train the membership function parameters to emulate the training data. There are two different optimization methods: hybrid or backpropagation. The number of training epochs and the training error tolerance are to

set the stopping criteria for training. The training process stops whenever the maximum epoch number is reached or the training error goal is achieved.

Figure 4.5.3-5 shows the ANFIS editor window with:

- *Training data (ooo)* - (top centre) shows the data (blue coloured) loaded for training the FIS system with x-axis number of data samples and y-axis sample values.
- *Load data* - (middle left) shows that the training data is loaded form file.
- *Generate FIS* - *(*middle centre*)* shows the grid partition has been selected as the clustering technique on the loaded training data.
- *Train FIS* - (middle right) shows the hybrid method with 0 error tolerance and 3 epochs is used for the training of FIS system.
- *Test FIS* - (middle right extreme) shows the training data is selected for the FIS system to train according to the data loaded.

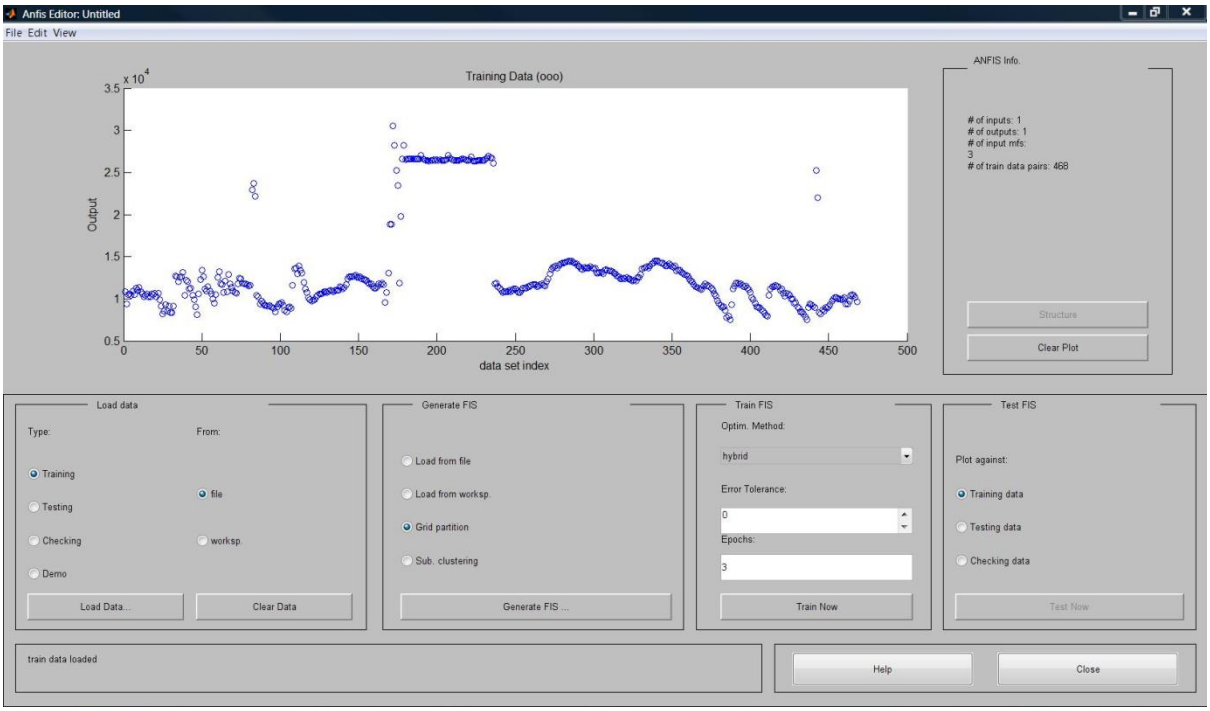


Figure 4.5.3-5. The training data loaded in the ANFIS structure (blue circles).

Figure 4.5.3-6 shows the newly created FIS structure ‘patient test’ in the ANFIS editor window with:

- *Training data(ooo)* - (top centre) shows the data (blue coloured) loaded for training the FIS system with x-axis number of data samples and y-axis sample values and FIS output in red coloured stars.
- *Anfis info* – (top right) shows the system info as 3 inputs (ECG, HR, BP), 1 output (Hypovolaemia) and 3 3 3 MFs (3 for each input).
- *Load data* - (middle left) shows that the training data is loaded from file.
- *Generate FIS* - *(*middle centre*)* shows the Subtractive clustering has been selected as the clustering technique on the loaded training data.
- *Train FIS* - (middle right) shows the Backpropagation method with 0 error tolerance and 200 epochs is used for the training of FIS system.
- *Test FIS* - (middle right extreme) shows the testing data is selected for the FIS system to test the FIS system according to the data loaded.

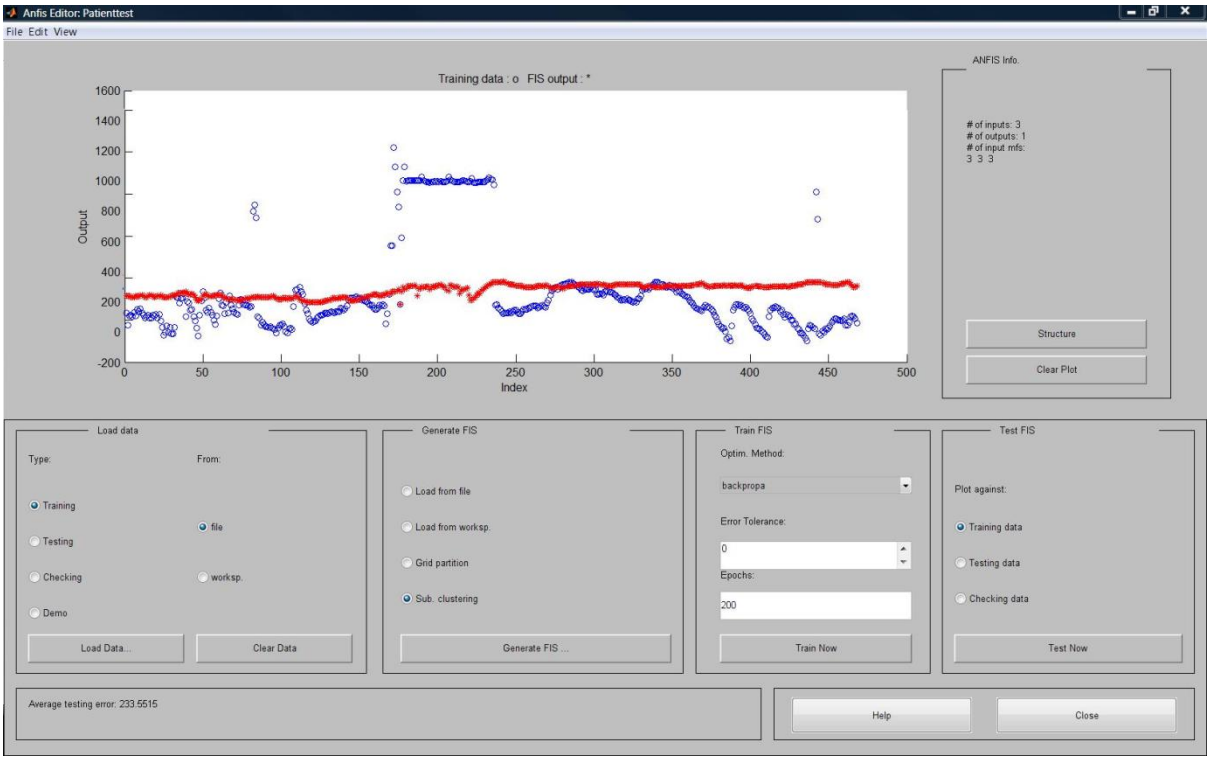


Figure 4.5.3-6. The training data against the FIS output (red stars).

4.5.3.4 Validating the Trained FIS

After the FIS is trained, validation of the model is carried out using **testing** or **checking** data that differs from the one we used to train the FIS. The figures below show the testing data with the tested FIS output.

Figure 4.5.3-7 shows the created FIS structure 'patient test' in the ANFIS editor window;

- *Testing data (. . .)* - (top centre) shows the data (blue coloured) loaded from trained data to test the FIS system.
- *Anfis info* – (top right) shows the system info as 3 inputs (ECG, HR, BP), 1 output (Hypovolaemia), 3 3 3 MFs (3 for each input) and *468 test data pairs created*.
- *Load data* - (middle left) shows that the testing data is loaded from file.
- *Generate FIS - (*middle centre*)* shows the Grid partition has been selected as the clustering technique on the loaded testing data.
- *Train FIS* - (middle right) shows the Backpropagation method with 0 error tolerance and 200 epochs is used for the training of FIS system.
- *Test FIS* - (middle right extreme) shows the testing data is selected to test the FIS system according to the trained data.

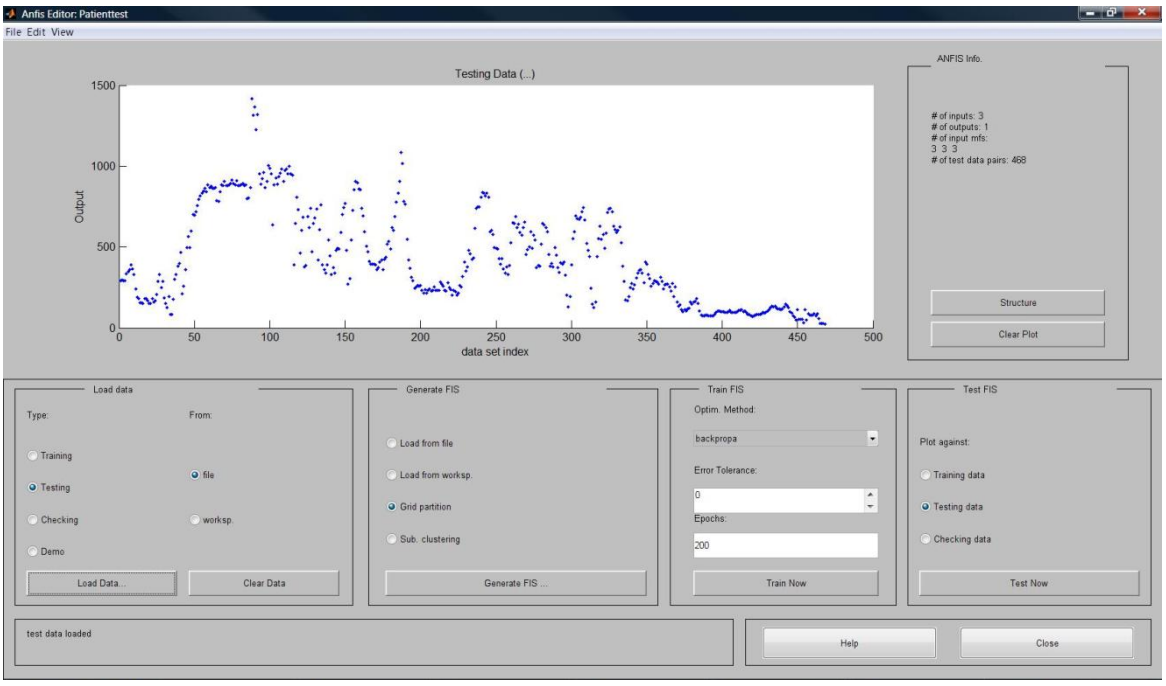


Figure 4.5.3-7. The testing data is shown (blue).

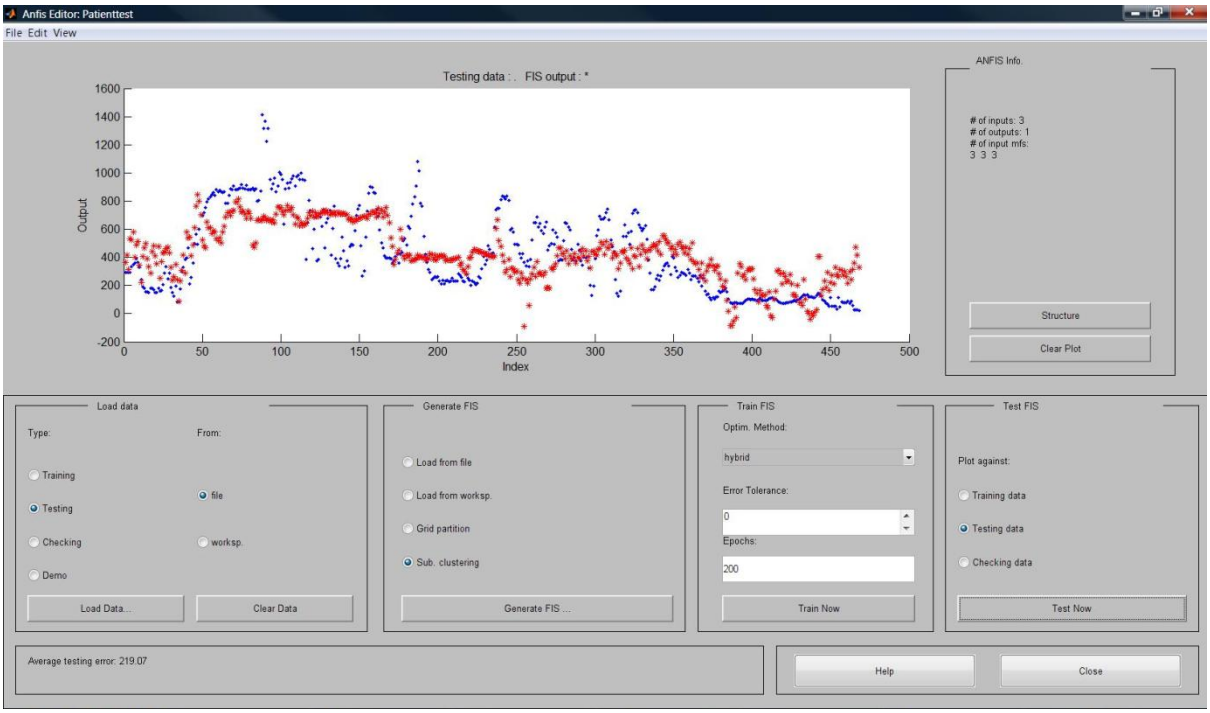


Figure 4.5.3-8. The testing data (in blue) is plotted against the FIS output (in red).

This section explained the FIS and ANFIS system using the Sugeno-type FIS structure. This includes creating, generating, training, testing, and validation of the FIS and ANFIS structure. The next section describes the Mamdani type fuzzy control system, which is the other method used in this fuzzy logic system. The basic idea and fundamental explanation of this Mamdani type of system is explained in section 4.3 of this chapter.

4.6 Mamdani Type Fuzzy Logic Monitoring System (FLMS)

4.6.1 The FIS Editor

Using the FIS editor, the Mamdani-type inference system is created with the name 'Patient Monitoring'. Using three inputs: ECG-HR, BP, and PV, and one output as Hypovolaemia.

The FIS Editor displays general information about a fuzzy inference system as shown in Figure 4.6.1-1:

- On the left side of the figure, the 3 boxes are the 3 inputs as HR, BP and PV which is mapped to the Mamdani type FIS structure named as 'Patient Monitoring'; this is mapped to the single created output named as

'Hypovolaemia'. These boxes are the pop-up menus that allow modifying of various pieces of the inference process.

- On the right side, at the bottom of the figure, is the area that displays the name of an input or output variable, its associated membership function type, and its range.
- At the bottom is a status line that relays information about the system.

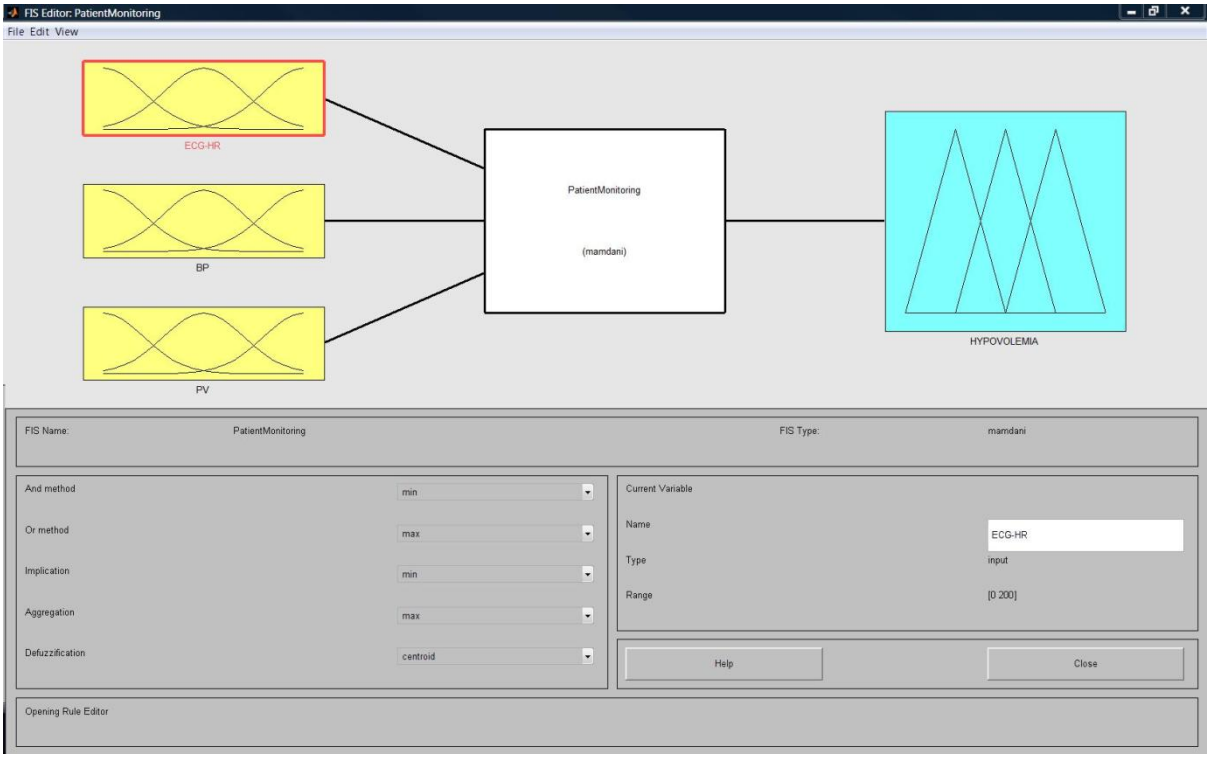


Figure 4.6.1-1. The FIS editor window of FLMS loaded with the Patient Monitoring FIS structure.

4.6.2 The Membership Function Editor

The Membership Function Editor is the tool that displays and edits all of the membership functions associated with all of the input and output variables for the entire fuzzy inference system. The membership functions from the current variable are displayed in the main graph. The ranges of the MFs are set after analysing the offline data of all the patients; this is explained in the starting sections of Chapter 3.

This is a very important step for building a solid system. If MFs are accurate, it will affect the whole performance of the system. The Membership Function Editor shares

some features with the FIS Editor, such as menu options, status lines, and Help and Close buttons.

Figure 4.6.2-1 is the membership function editor window of Patient Monitoring FIS structure, and shows:

- *FIS variables* – (top left side) shows the three input and one output variables of this structure.
- *Membership function plots* – (top center) shows the MF structure of the first input ECG which is highlighted on the left and also shows the three functions for each input as mild, moderate, and severe.
- *Current variable* – (middle left) shows the details of the selected input (ECG) as name, type, range, and display range.
- *Current membership function* – (middle right) shows the details of the selected membership function (mild) of the 1st input as name, type, parameters.

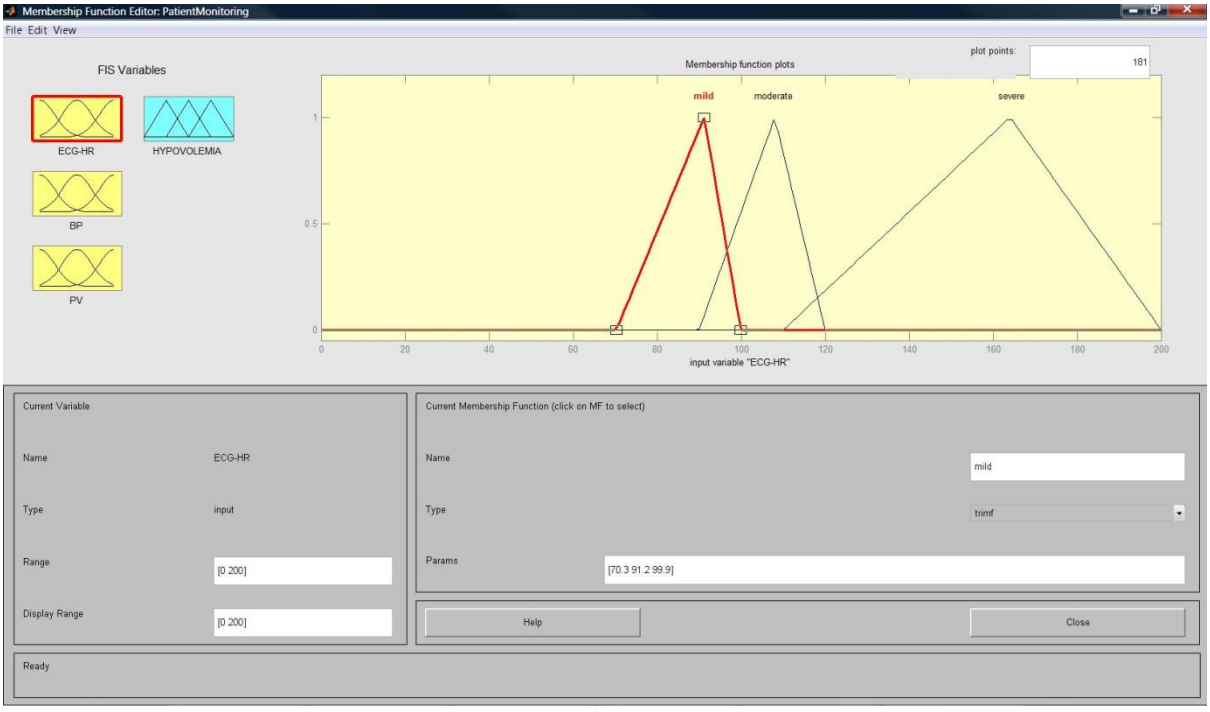


Figure 4.6.2-1. Membership function editor window of FLMS.

4.6.3 The Rule Editor

Constructing rules using the graphical Rule Editor interface is fairly self evident. Based on the descriptions of the input and output variables defined with the FIS Editor, the Rule Editor allows the programmer to construct the rule statements automatically.

- From the GUI, create rules by selecting an item in each input and output variable box and one *connection* item and clicking *add rule*.
- Choose *none* as one of the variable qualities to exclude that variable from a given rule and choose *not* under any variable name, to negate the associated quality.
- Delete a rule by selecting the rule and clicking *delete rule*.
- Edit a rule by changing the selection in the variable box and clicking *change rule*.
- Specify weight to a rule by typing in a desired number between 0 and 1 in *weight*.

Similar to those in the FIS Editor and the Membership Function Editor, the Rule Editor has the menu bar and the status line.
Figure 4.6.3-1 is the rules editor window of Patient Monitoring FIS structure, it shows:

- Top - seven different rules using the three inputs and one output with the rule weigh as (1) can be seen at the end of each rule.
- Middle – three input MF windows (ECG-HR, BP and PV) on the left and 1 output MF window (hypovolaemia) on the right.
- Bottom – Connection (on left) shows the AND operator and weight as 1 with different buttons as delete rule, add rule, and change rule.

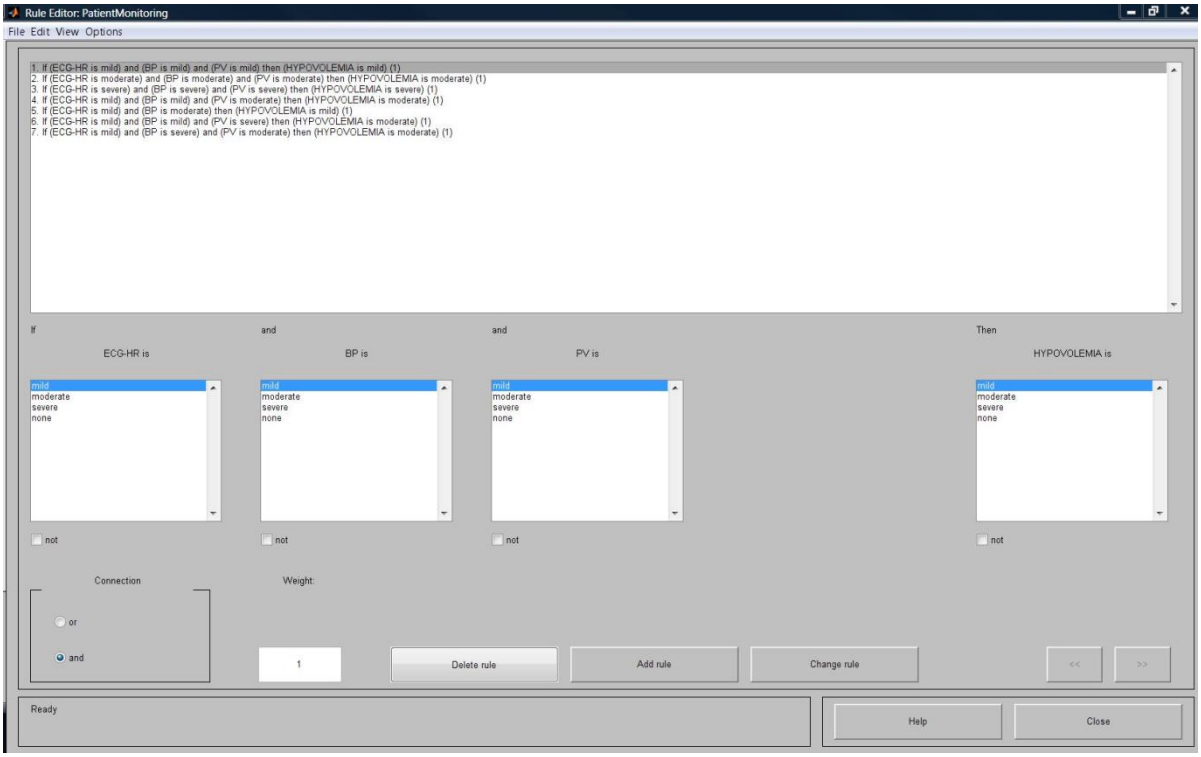


Figure 4.6.3-1. Rule editor window of FLMS.

4.6.4 The Rule Viewer

The Rule Viewer displays a roadmap of the whole fuzzy inference process. Figure 4.6.4-1 there is a single figure window with plots nested in it. The plots across the top of the figure represent the antecedent and consequent of the first rule. Some of the plot information is given below.

- Each rule is a row of plots, and each column is a variable.
- The rule numbers are displayed on the left of each row.
- The first three columns of plots show the membership functions referenced by the antecedent, or the if-part of each rule.
- The last (fourth) column of plots shows the membership functions referenced by the consequent, or the then-part of each rule.
- The last plot in the third column of plots represents the aggregate weighted decision for the given inference system.
- The defuzzified output is displayed as a bold vertical line on this plot.

Figure 4.6.4-1 is the rules viewer window of Patient Monitoring FIS structure, it shows:

- Column 1 ECG-HR - 7 different rule boxes using the 1st input.
- Column 2 BP - 7 different rule boxes using the 2nd input.
- Column 3 PV - 7 different rule boxes using the 3rd input.
- Column 4 HYPOVOLAEMIA – 7 different rule boxes using all three inputs.
- Column 4 (bottom box) – is the output rule result box with the red line shows the value of output at the given input.

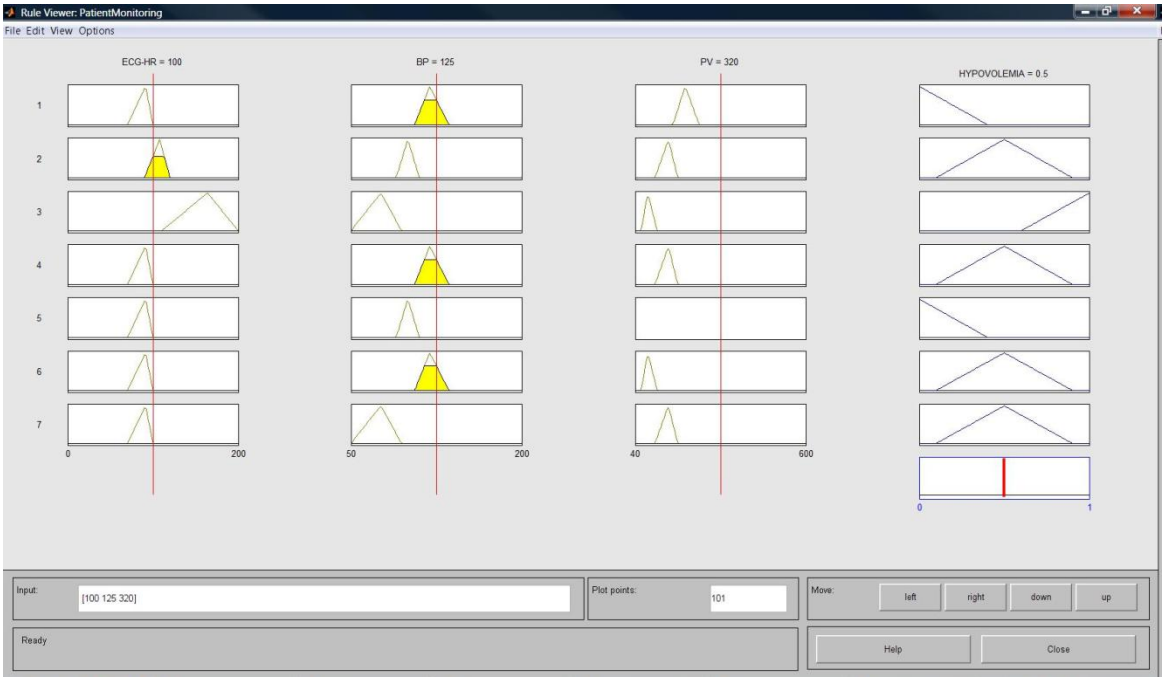


Figure 4.6.4-1. Rule viewer window of FLMS.

4.6.5 Tuning and System Enhancement

Tuning of the monitoring system is done by changing the rule antecedents, changing the centres of the input and/or output membership functions, adding additional degrees to the input and/or output functions such as ‘Mild’, ‘Moderate’, and ‘Severe’ levels and ‘Hypovolaemia’ as output response of the system. These new levels generate additional rules and membership functions which overlap with adjacent functions, forming longer ‘mountain ranges’ of functions and responses. The different techniques used for making these changes systematically are one of the turning points for the work of this project.

The logical product of each rule is inferred to arrive at a combined magnitude for each output membership function. Once inferred, the magnitudes are mapped into their respective output membership functions, delineating all, or part, of them. The ‘fuzzy centroid’ of the composite area of the member functions is computed and the final result taken as the crisp output. Tuning the system amounts to ‘tweaking’ the rules and membership function definition parameters to achieve acceptable system response.

4.7 Summary

Fuzzy Logic provides a completely different, unorthodox way to approach this anaesthesia monitoring system. This method focuses on what the system should do rather than trying to understand how it works. One can concentrate on solving the problem rather than trying to model the system mathematically, if that is even possible. This almost invariably leads to quicker and better solutions. Once understood, this technology is not difficult to apply and the results are usually quite surprising and pleasing.

A fuzzy logic monitoring system has been developed and tested successfully. Before the final development phase each MF, the rules and structure, were checked several times with necessary changes according to its performance. The results are discussed in the next chapter and are compared with the results from the SCADA monitoring system. There is also a comparison of results from previous work done by Gohil [87] with his RT-SAAM.

CHAPTER 5 Overviews, Testing and Results

5.1 Introduction

This project seeks to develop a new monitoring system for representing the abnormalities in physiologic variables, to enhance a clinician's ability to see and rapidly respond to critical events. A digital visualization of an individual's physiologic data is also created. It is both a probing and a representational system that brings together science and art through architectural design. The project's goal was to create a coordinated and interactive hyper-representation, so that it articulates the physiological data in a format that is easily and quickly understood by the clinician. Raw data is obtained from the existing medical equipment that measures human physiological signs.

Using this data, 3-D objects that represent physiologic changes within the body are created in digital space to show functional relationships that aid in the detection, diagnosis, and treatment of critical events. It has already been stated that the domain of anaesthesia suffers from a high degree of complexity [120]. A representation of this knowledge that maximises its availability can therefore be expected to play an important part in determining the performance of the monitoring system.

The clinician’s satisfaction and trust in a particular system are essential to the success of any practical expert monitoring system; it is therefore essential that the system be programmed to display explanations to the clinician of all the reasoning processes; additionally, the encoded expert knowledge must evolve over time, while being continually sanctioned by the professionals.

There is clearly a need to be able to represent knowledge in more than one form in order to produce comprehensive monitoring [121]. For this project, it was decided to concentrate research mainly into two completely different techniques/methods. The diagnoses to be made are therefore influenced by a more limited range of knowledge, for which a particular representation (SCADA and Fuzzy Logic) was used, as will be discussed later.

In this chapter there will be the discussion on two different techniques (which are earlier explained in detail in Chapter-3 (SCADA) and Chapter4 (fuzzy logic) :

- SCADA Monitoring System (SMS) – The anaesthesia monitoring system developed using the SCADA software called InTouch. This system uses its own built-in tools and algorithm language for communication with protocols, inputs, outputs, etc. The next section describes the system results.
- Fuzzy Logic Monitoring System (FLMS) – The anaesthesia monitoring system developed using MATLAB software and its fuzzy logic toolbox. This system uses Mamdani type of FIS structure with MF's, rules, sets, etc. The system results are discussed in a later section of this chapter.

5.2 SCADA Monitoring System (SMS)

The current version of SMS is designed to diagnose one specific condition – hypovolaemia. This condition was selected because it is one of the important and critical occurring events to be identified; failing to do so may have serious consequences. The information about the situation associated with each patient's diagnosis and its consequences reported in medical literature is a knowledge requirement. This knowledge was collected from anaesthetists, anaesthetic texts, medical literature, and other sources.

In the course of developing the SMS, few problems have been found. This is essentially an issue of completeness of the information on which a diagnosis is based, and as such is handled with more vigilance. In all, SMS makes use of features extracted from three signals (HR, BP, and PV). The linguistic terms representing the symptom is extracted from these signals.

5.2.1 System Design Overview

This section gives an idea of SMS's overall design with an overview.

Figure 5.2.1-1 shows the building blocks of the complete system. It is classified into six main sections. The top three boxes are explained in detail as the pre-processing part of this project in Chapter 2 and the bottom three boxes are explained as the development part of SCADA system in Chapter 3.

1. *Raw Data* – The raw patient data was collected by an anaesthesiologist at Auckland Hospital with informed consents. It's in the Excel data files which are converted through the DOMonitor application discussed in detail in Chapter 2.
2. *Filtering* – A number of different filtering techniques were applied to reduce the noise, and major artefacts were removed for the diagnosis process.
3. *Filtered Data* – After filtering, the data is given to the SMS structure for diagnosing.
4. *Diagnosis* – This has been done using the filtered data of 15 patients selected randomly from the total of 30 patients' data.
5. *Testing* – Using the above-mentioned three conditions for the generation of warning, the system has been tested with beneficial and positive results, which are discussed in the coming section.
6. *Warning/Alert* – This is the output where the system gives the warning or an alert for the level of hypovolaemia.

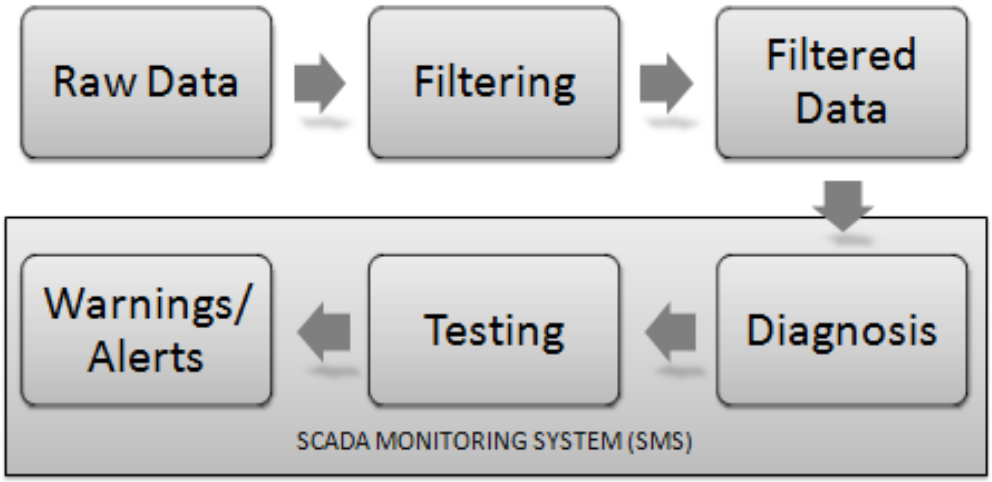


Figure 5.2.1-1. Block diagram of the complete design structure.

5.2.2 SMS Structure Overview

This section summarises the overview of modules used and major contents of the SMS system developed during the project work. Covering the purpose and reasoning behind the system's components always provides more strength.

Figure 5.2.2-1 shows the final implementation of SMS which can be divided into seven main sections. Each one of these conditions and modules is explained in Chapter 3.

1. *AM, WM, &WV* – The application manager, window maker, and window viewer; these modules are used for the creation of the SMS as a system.
2. *GUI Display* – The graphical user interface display panel which is used as the main communication with the clinician.
3. *Tags* – One of the important phases of development was assigning the correct tag to appropriate mapped variable, according to its I/O, memory, alarm, and trend.
4. *RT-Alarms* – Real-time alarms are the backbone of the SMS. The SMS system depends on the output of the RT-Alarms warnings and alerts.
5. *HG-Trends* – Historical graphical trend is one of the important modules used in this system development.
6. *Animations* – This links to all variables in the GUI and system to perform in the display front panel.
7. *Conditions* – Three different conditions are set for the detection of Hypovolaemia in this system.

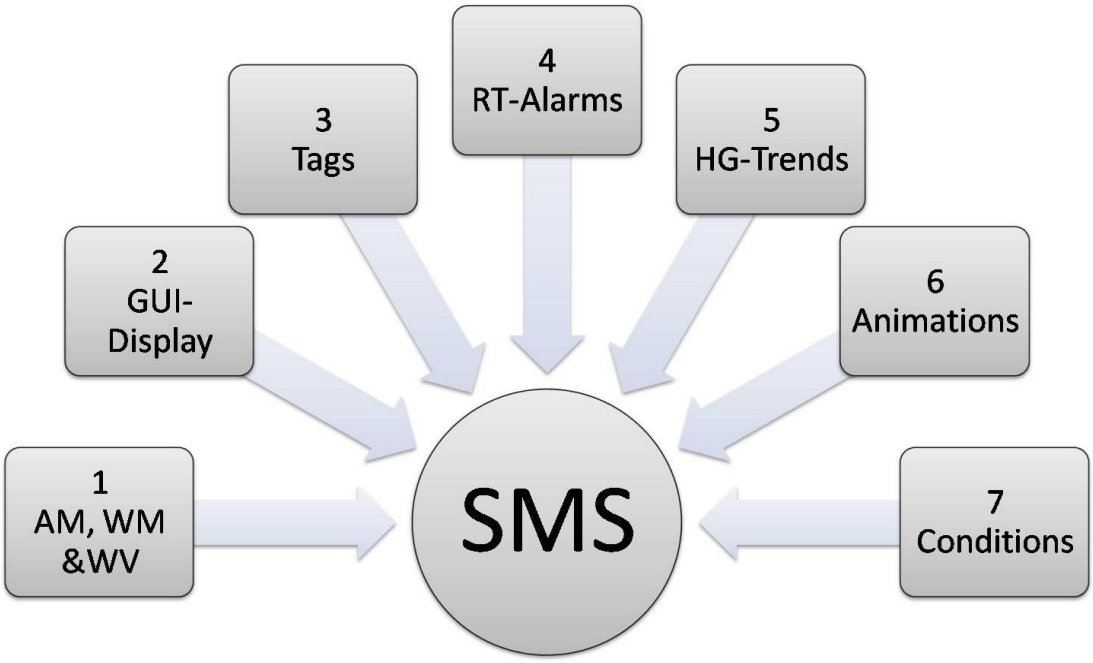


Figure 5.2.2-1. Block diagram for the general overview of the SMS with its seven major contents where AM is application manager, WM is window maker, WV is window viewer, GUI is graphical user interface, RT is real time, HG is historical graphical.

5.2.3 SMS Flowchart

This section shows the SMS flow chart. Figure 5.2.3-1 shows the step-by-step procedure of the SMS structure. The following items describe the flow chart diagram.

1. *Raw Data* – Patient data which contains noise and artefacts needs to be pre-processed for better diagnosis and higher performance of the system.
2. *Filtering Techniques* – Raw data is filtered using the low pass filtering and the variance based filtering.
3. *Batch Process (5 minutes)* – The total data is divided into five minute intervals for better accuracy and fault detection. Each five-minute set of data was used as a batch and sent to the SMS for testing. Later, in some patients, the five-minute interval was increased to 10 minutes and further to 15 minutes, depending upon the data and its quality.
4. *C1: Condition 1: Threshold Limit* – Each physiological parameter is set with its own threshold limit. The threshold limit should be exceeded for each feature separately for the system to test for the second condition. If not, then return to step 3 (batch process) for next slot of 5-minutes' data.

5. *C2: Condition 2: Standard Deviation* – Each parameter should exceed its SD limit (which is set after applying normalization method to all the offline data) for the system to test for the third condition. If not, then return to step 3 (batch process) for next slot of 5-minutes' data.
6. *C3: Condition 3: Condition 1 + Condition 2* – The system checks whether each parameter is true in its normal threshold values (condition-1) and each parameter exceeds the SD limit (condition-2). If both conditions are true then condition-3 is true for the system to give the output. If not, then return to step 3 (batch process) for next slot of 5-minutes data.
7. *Warnings* – If these three conditions test true, then the system would generate the output as the warning for hypovolaemia in the patient and communicate whether it is mild, moderate, or severe.
8. *Alerts* – The system gives the alerts if only condition 1 or condition 2 is true, but no alert for the condition 3; instead, it gives a warning.

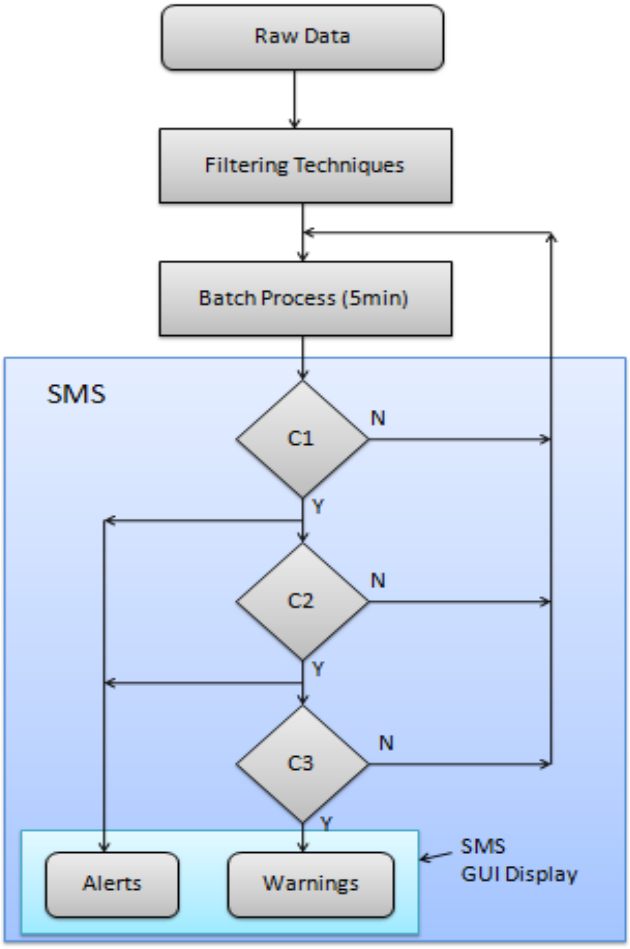


Figure 5.2.3-1. Flow chart of the SMS structure where C1 is condition1, C2 is condition2, C3 is condition3, Y is yes/true, N is no/false and SMS is SCADA monitoring system.

5.3 SMS Testing

5.3.1 SMS System Validation

SMS has been tested offline with data from 15 patients – a total of approximately 50 hours of data. Real-time performance of the system cannot be validated at this time because it has not been implemented in real-time mode, but it has the capability to work in real-time environment. The performance of SMS was validated offline as described in the next section.

It is estimated that more test sessions will be required for successful testing and refining of SMS. Initially 10 sets of patient data were taken as first phase of testing. The process was repeated several times for further refining the system. In the second phase of testing, five more patients' data were added. This has been selected randomly to analyse the system for fine tuning and to make SMS diagnose more accurately.

5.3.2 SMS Display Panel (GUI)

Figure 5.3.2-1 shows the front display panel of the SMS. There are four important features to be discussed, as marked with the numbers in blue. They are:

1. Title: It shows "INTOUCH-WINDOWVIEWER-C:\DOCUMENTS AND SETTINGS\MYDOCUMENTS\MY INTOUCH APPLICATIONS\PATIENT" this is the complete path of the system in the local memory where InTouch is the name of the SCADA software; Window Viewer is the output window, and rest is the location and system application name 'PATIENT'.
2. Alarm chart: This is the real-time alarm chart which displays and saves the warning and alerts. The different columns are: date, time, status, class, type, priority, name, group, provider, value, and limit. At the bottom of the alarm chart is the message space which shows the update and active state of the alarm chart.
3. Historical graphical trend: This displays the graphical value of different input physiological parameters at one time in the real-time mode. It is the real-time

historical graphical trend display. X-axis displays the value of the parameter and Y-axis displays the time in every one second interval (this can be changed).

4. Digital report: This is the digital display of the parameters which are of interest to the user.

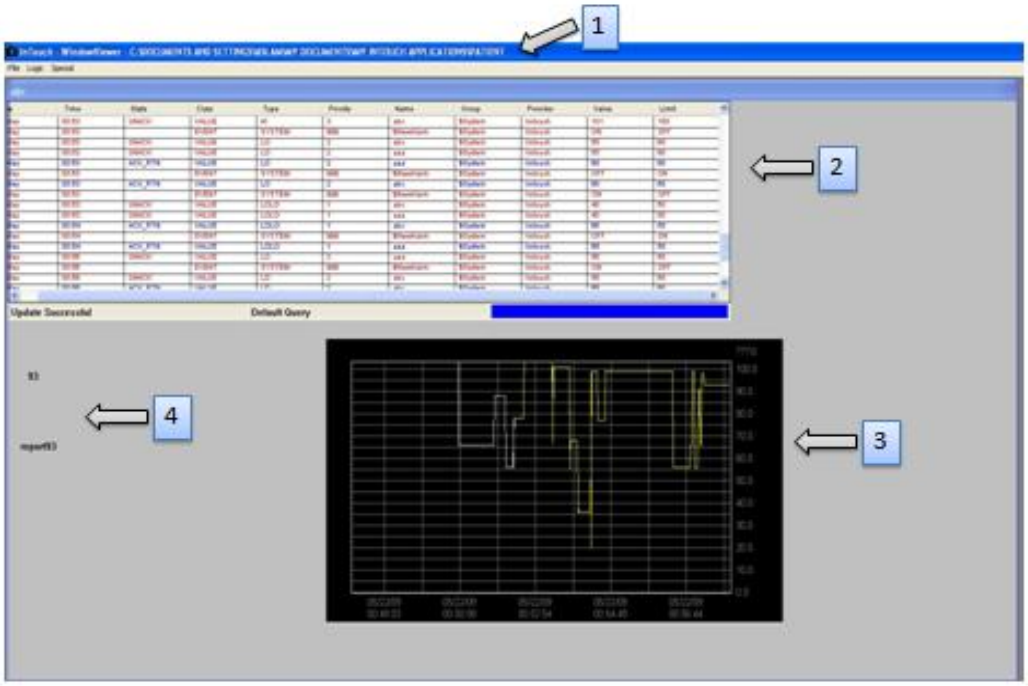


Figure 5.3.2-1. Front display panel of SMS.

5.3.2.1 Addition to the Display Panel of SMS

In the final development stage of SMS the power tool was added to the front display panel because it shows the minute details of any particular parameter, which may be of more interest to the clinician at that time. This is an easy to handle and user-friendly tool. This power tool shows the real time value, alarm display, percentage change, digital value, runtime status, and properties of the selected feature. This power tool is added into the GUI panel of the system in the final development mode of SMS.

Figure 5.3.2-2 shows the power tool of the SMS;

- *Feature Name* – The name of the parameter that needs to be monitored closely (ECG-HR in this case) should be entered in the relevant space of the window only in the run mode of the SMS. The first three characters will be enough.

- *Real-time value* – It shows the real-time value of the parameter in digital format.
- *Alarm Display* – Shows the warning/alarms of the particular parameter if it goes out of normal range, which is set by the developer so as to detect the main output hypovolaemia.
- *Limit Meter* – It shows the upper and the lower limit of that parameter.
- *Conditions Bar* – Shows the limits in the digital format as LL- lolo, L-lo, H-hi, HH-hihi. Under each limit there is the digital value which shows the value of the limit exceeded.
- *Properties bar* – (runtime values for each property) – Shows the changes in the parameter value in terms of percentage, MD%- percentage of major deviation, mD%- percentage of minor deviation and roC%- percentage of rate of change. Their runtime values are shown under each property.
- *Alarm and Deviation dead band limit* – This shows the value at which the alarm is to be generated, with the corresponding SD value limit shown as 20. This means that the alarm and SD value are connected to each other for the generation of alerts;

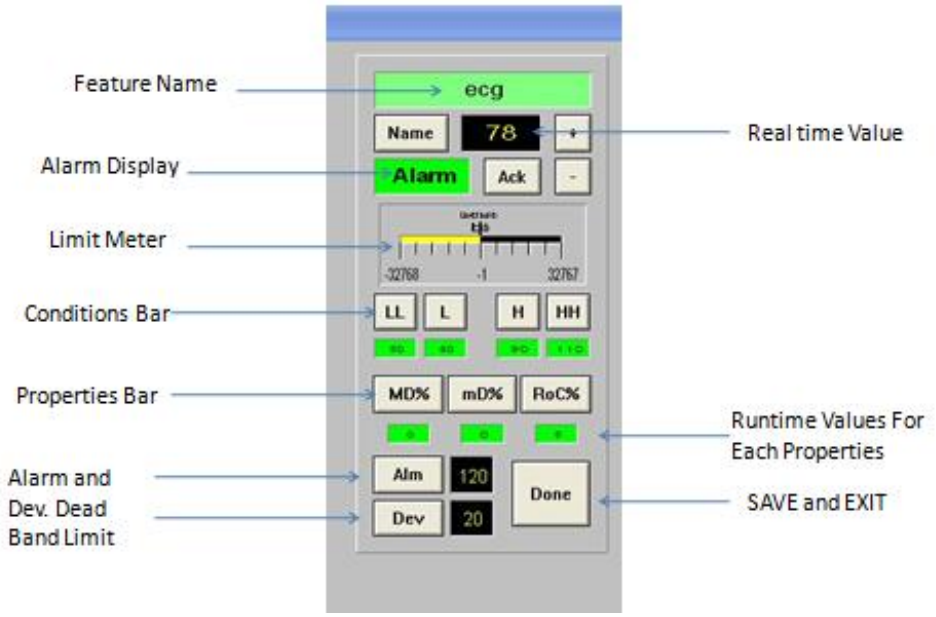


Figure 5.3.2-2. Power tool added to the display panel of the SMS.

5.3.3 SMS's Analysis, Evaluation and Feedback

Anaesthetists can record their response by clicking on the acknowledge button which is always next to the alarm display window of RT-Alarm chart in the front display panel. This button, shown as 'ACK' (acknowledge) means if the anaesthetists does not agree with the alert/warning generated by the SMS, then by simply pressing the ACK button the alert will be turned off until the next critical result occurs. This acknowledgment, which is given to the SMS by the anaesthetist, will be saved as the response and can be reviewed as the performance of the SMS.

The diagnostic alarm levels generated by various modules in SMS are evaluated for reliability by verifying them against the clinical evidence and the anaesthetist's response. This comparison is carried out for every 5-minute epoch. The level of agreement between the computer generated (SMS) diagnosis and the experts' (anaesthetists') diagnosis gives a performance indicator for the prototype diagnostic system.

Prior to proceeding to the next section it is imperative to acknowledge the fact that it is very likely that there might be some variability regarding the diagnoses among a group of anaesthetists [122, 123]. Experts handle each clinical event in real-time by monitoring the patient's hemodynamic state and making a therapeutic decision by forming a mental map of the available clinical evidence [124].

Diagnosis of any critical events during anaesthesia depends on all available information (evidence) and the ability of the expert to recognise patterns of disease behaviour. Expert diagnosis may vary between anaesthetists, and authenticity of the diagnosis depends on their skills and experience. Thus there is nothing like 100 percent true or 100 percent false diagnosis, as even the anaesthetists' diagnoses have uncertainty attached to them [43]. Thus, for evaluating SMS's diagnostic performance, the level of agreement between SMS and the anaesthetists is used as a measure of how accurately SMS can help anaesthetists' performance. The following section shows the diagnosis results of SMS.

5.3.4 SMS Diagnosis

The patients' data sets had incomplete expert diagnoses (anaesthetists' diagnoses) for some of the 15-minute epochs; expert diagnoses were not available. Dr. Harrison studied the clinical evidence for each patient and generated the corresponding diagnoses for the incomplete information. All the clinical events and procedures that took place during each of these data collection sessions were saved into text files. These text files provide the clinical evidence which could be used by an expert for retrospectively analysing each 15-minute epoch. A sample log file and the corresponding physiological data can be found in Appendix A.

Even though retrospective expert diagnosis cannot be as accurate as the real-time diagnosis given by the anaesthetist, it may be considered very close to the real-time diagnosis. Expert opinion in anaesthesia varies between anaesthetists, and there is no gold standard for expert diagnosis. The accuracy of the expert diagnosis in case of anaesthesia-related critical events depends totally on the expertise of the anaesthetist.

The SMS offline version of the alarm system is capable of reading the given patient data files; it analyses the data in these files and generates the alarm levels for the entire length of the data-set. The data from the digital patient files was fed to SMS, and the corresponding computer-generated diagnostic alarm levels were produced by three classified outputs as mild, moderate, and severe.

The SMS front graphical display shows the output result in the form of alarm, trend, lights, and digital report (values). The following rules were used and applied to the SMS for the generation of alerts for the given subset of data (5 min) and the main set of data (15 minutes).

5.3.4.1 Condition – 1 Crisp Numerical Values

Table 5.3.4-1 shows the testing limits set for condition-1 in SMS for mild, moderate, and severe (hypovolaemia). These limits are the crisp values and were set after analysing a number of patients' data in those three physiological parameters, and theses crisp value ranges are set for the condition-1.

1. *Heart Rate* – In case of Hypovolaemia, HR will increase.
    1.1. Mild – It is set from 90 BPM to 100 BPM.

1.2. Moderate – It is set between 100 BPM to 120 BPM.

1.3. Severe – It is set from 120 BPM and above as severe Hypovolaemia (condition-1).

2. *Blood Pressure* – In case of Hypovolaemia, BP will decrease or fall.

2.1. Mild – It is set from 100 mm Hg to 90 mm Hg, as we are only considering low limits (fall in BP) of this feature.

2.2. Moderate – It is set between 90 mm Hg to 80 mmHg.

2.3. Severe – It is set from 80 mm Hg and less as severe Hypovolaemia.

3. *Pulse Volume* – In case of hypovolaemia PV will decrease or fall.

3.1. Mild – It is set from 120 to 100, as we are only considering the fall in PV of this feature.

3.2. Moderate – It is set from 100 to 80.

3.3. Severe – It is set from 80 and less, as severe Hypovolaemia.

Table 5.3.4-1. Condition-1 testing values and limits for SMS.

HYPOVOLEMIA	MILD	MODERATE	SEVERE
Heart Rate ⇧	90 – 100	100 – 120	120 & >
Blood Pressure ⇩	100 – 90	90 – 80	80 & <
Pulse Volume ⇩	120 – 100	100 – 80	80 & <

5.3.4.2 Condition – 2 SD Changes

Table 5.3.4-2 shows the testing Standard Deviation (SD) limits set for condition-2 in the SMS for mild, moderate, and severe (Hypovolaemia). These limits are the SD changes with respect to its original values and were set after analysing a number of patients' data in those three physiological parameters and SD limits range are set for condition-2.

1. *Heart Rate* –

1.1. Mild – It has been set from 2.75SD – 4SD.

1.2. Moderate – It is set between 4SD – 6SD.

1.3. Severe – It is set from 6SD and above as severe Hypovolaemia.

2. *Blood Pressure* –

2.1. Mild – It has been set from 2.5SD – 4SD.

2.2. Moderate – It is set between 4SD – 6SD.

2.3. Severe – It is set from 6SD and above as severe Hypovolaemia.

3. *Pulse Volume* –

3.1. Mild – It has been set from 2.SD – 4SD.

3.2. Moderate – It is set from 4SD – 6SD.

3.3. Severe – It is set from 6SD and above as severe Hypovolaemia (condition-2).

Table 5.3.4-2. Condition-2 testing SD values and limits for SMS.

HYPOVOLAEMIA	MILD	MODERATE	SEVERE
Heart Rate (SD)	2.75 – 4	4 – 6	6 & >
Blood Pressure (SD)	2.5 – 4	4 – 6	6 & >
Pulse Volume (SD)	2 – 4	4 – 6	6 & >

5.4 SMS's Results

Based on these classification rules, each of the patient's data has been monitored and a comparison chart constructed.

5.4.1 SMS's Analysis for Patient Number 11

Table 5.4.1-1 shows a sample comparison chart for Patient Number 11. Similar comparison charts were constructed for all 15 patients. Based on the agreement between the experts' diagnosis and SMS's diagnosis, each of these was classified into three categories - mild, moderate, and severe. Some important points for better understanding of the results and their comparison with the anaesthetist's response are given below:

- Column 1 – Time 15 minutes: Slot is presented in the table to match the anaesthetist's diagnosis of 15-min time interval, also sub time slots of 5 minutes were analysed in the background for SMS diagnosis.

- Column 2 – Anaesthetist's diagnosis: This diagnosis was provided by Dr. Harrison, this is taken as the comparison and performance review of the SMS.
- Column 3 – SMS Result: In the output, Hypovolaemia is classified into mild, moderate, and severe.

The blank spaces in the table are the situations where no alert/ warning have occurred or neglected due to incomplete data.

Table 5.4.1-1. SMS diagnosis result (for patient #11).

Patient – 11 Filename – 140408 (In record it is '**Pat11 140408**')				
Time 15 minutes	Anaesthetist's diagnosis	SCADA Monitoring system (SMS)		
		MILD	MODERATE	SEVERE
0800				
0815				
0830	N	N	N	N
0845	N	N	N	N
0900	N	N	N	N
0915	N	N	N	N
0930	N	N	N	N
0945	N	N	N	N
1000	N	N	N	N
1015	N		P	
1030	P			P
1045	P			P
1100	P			P
1115	P			P
1130	P		P	
1145	P		P	
1200	P		P	
1215	P	P		
1230	N	P		
1245	N	N	N	N
1300				

Where, N = No, P = Possible in the table.

5.4.2 SMS Diagnosis vs Anaesthetist's Diagnosis and Comments

During collection of the data the attending anaesthetist can make annotations regarding any significant events. These annotations are then compared to the diagnoses of SMS obtained off-line at a later date. The data were also examined for any significant events

not annotated at the time by the anaesthetist. These sets of events provided justification for any alarms or warnings generated by SMS. These annotations were saved as a text file, with the patient's data file for supporting reference and could then be compared to the diagnoses of SMS. The data was also examined for any significant events not annotated at the time by the anaesthetist. These sets of events provided justification for any alarms or warnings generated by SMS. Sample patients' log file and data file are in Appendix B.

The table below shows the agreement between the SMS offline results and the expert's comment based on the agreement between the expert's diagnosis and SMS's diagnosis; the complete set of data was classified into four categories (TruePOS, TrueNEG, FalsePOS and FalseNEG). This type of analysis was done to test the SMS overall result and performance with the other systems.

SMS results are compared with another similar monitoring system called SAAM [32, 75, 87]. In that system, the expert (anaesthetist) and SAAM were the two observers interpreting the diagnoses for the pathological events from the physiological data. Kappa analysis was used to compare the results of SAAM. The value given from the Kappa analysis computed indicates the level of agreement/disagreement between the expert and SAAM.

In the next section, the results of SMS will be analysed through the Kappa analysis and compared with the SAAM's results. The SMS and SAAM are both anaesthesia monitoring systems, but use different techniques and methods for detecting the anaesthesia-related events.

Table 5.4.2-1. Patient – 11 Filename – 140408 (In record it's 'Pat11 140408')									
Time 15 minutes	Anaesthetist's diagnosis	SCADA Monitoring system (SMS)			Anaesthetist's Comments	Agreement		Disagreement	
		MILD	MODERATE	SEVERE		True POS SMS +ve Expert +ve	True NEG SMS -ve Expert -ve	False POS SMS +ve Expert -ve	False NEG SMS -ve Expert +ve
0800									
0815					Induction 0824 (intubated 8:28)				
0830	N	N	N	N	Still setting up		1		
0845	N	N	N	N	Still setting up		1		
0900	N	N	N	N	Central line in		1		
0915	N	N	N	N	-		1		
0930	N	N	N	N	Incision		1		
0945	N	N	N	N	-		1		
1000	N	N	N	N	Dissection CVP 4		1		
1015	N		P		Dissection CVP 4			1	
1030	P			P	Dissection CVP 4	1			
1045	P			P		1			
1100	P			P		1			
1115	P			P	Morphine given	1			
1130	P		P			1			
1145	P		P		Volume loading started 11:50	1			
1200	P		P		Volume loading continuing	1			
1215	P	P			Volume loading continuing	1			
1230	N	P						1	
1245	N	N	N	N			1		
1300									
TOTAL						8	8	2	0

The first column (time15 minutes) of the Table 5.4.2-1 shows the two time slots in red colour, that is:

- 10:15 – Shows the expert diagnosis N (no) and SMS's result shows that there is the moderate level of hypovolaemia. From the time slot 10:15 to 10:30 there may be some possibility of Hypovolaemia because the expert's assessment is of last 15 minute slot. It goes in the cycle of a 15-minutes time interval. As you can see, there is P (possible) shown by the expert in the next time interval of 10:30 to 10:45. SMS's diagnosis detected moderate hypovolaemia in Patient Number 11 at 10:24, and the warning was generated.
- 12:30 – The expert's diagnosis is N (no) and SMS's result shows that there is a mild level of hypovolaemia. From the time slot 12:30 to 12:45 there may be some possibility of inconsistency because the expert's assessment is of the last 15-minute slot, that is, from 12:15 to 12:30 which shows the P (possible). SMS's diagnosis detects the mild level of hypovolaemia in Patient Number 11 at 12:32, and the warning is generated.

For further analysis of the system's performance, these are some of the issues which need explanation by the expert. Hence, in Chapter 6 which follows, there will be a detailed discussion on the outcome of SMS. The next section provides a description of results from the SMS monitoring system and also comparison of the results with other expert systems.

5.4.3 SMS's Result analysis

This section will demonstrate the efficiency and check the performance of the SMS system. To test the level of agreement between the anaesthetists and the SMS, Kappa analysis was used (see Kundel and Polansky [125]). Using this Kappa analysis Gohil [87] tested the level of agreement between his RT-SAAM and the experts' diagnoses. Kappa gives a statistical measure for evaluating the agreement and shows how often two or more observers agree/disagree in their opinions.

5.4.4 SMS - Kappa Analysis for Patient Number 11

For computing the level of agreement between the anesthetist's diagnosis and SMS's diagnosis, the output was divided into four classes. Based on the positive or negative

diagnosis generated by SMS and the diagnosis by the anesthetists, there were four possible permutations for hypovolaemia diagnosis:

a) Both SMS and Expert agree that hypovolaemia exists (TruePOS).
b) Both SMS and Expert agree that hypovolaemia does not exist (TrueNEG).
c) SMS gives positive diagnosis while Expert gives negative diagnosis (FalsePOS).
d) SMS gives negative diagnosis while Expert gives positive diagnosis (FalseNEG).

Using these classifications, the sample Kappa analysis for Patient Number 11 is shown below. From the Table 5.4.2-1 the values of TruePOS, TrueNEG, FalsePOS and FalseNEG have been taken.

Table 5.4.4-1. Kappa table for Patient Number 11.

	Expert (+ve)	Expert(-ve)	Total
SMS (+ve)	8	2	10
SMS (-ve)	0	8	8
Total	8	10	18

Based on the data from Table 5.4.4-1the sample Kappa analysis is carried out for Patient Number 11 using equations 5.1 and 5.2. The sample calculation shows the positive agreement (P_{pos}) and negative agreement (P_{neg}) indices were calculated as follows.

$$P_{pos} = \frac{8+8}{(8+2)+(8+0)} = 0.89 \qquad \textbf{5.1}$$

$$P_{neg} = \frac{8+8}{(2+8)+(0+8)} = 0.89 \qquad \textbf{5.2}$$

The third index of agreement gives the overall agreement (P_o) level between the expert and SMS.

$$P_o = \frac{8+8}{18} = 0.89 \quad \textbf{5.3}$$

$$P_e = \left(\frac{10}{18}.\frac{8}{18}\right) + \left(\frac{8}{18}.\frac{10}{18}\right) = 0.49 \quad \textbf{5.4}$$

Agreements between the two diagnoses may be affected by chance. Kappa (k) is a measurement of agreement between the expert and SMS which has been corrected for error by chance. Kappa (k) is calculated by subtracting the proportion of readings that are expected to agree by chance (P_e) from the overall agreement (P_o) and dividing the remainder by the number of cases on which agreement is not expected to occur by chance.

$$K = \frac{P_o - P_e}{(1-P_e)} = 0.78 \quad \textbf{5.5}$$

The standard error (SE) of k for a 2x2 table could be estimated with Eq.5.6.

$$SE = \sqrt{\frac{P_o(1-P_o)}{n(1-P_e)^2}} \quad \textbf{5.6}$$

$$SE = \sqrt{\frac{0.89(1-0.89)}{18(1-0.49)^2}} = 0.145 \quad \textbf{5.7}$$

The 95% confidence intervals (CIs) for k could be calculated with the following equation.

$$CI_{95\%} = K \pm 1.96 \times SE \quad \textbf{5.8}$$

CIs for k were

a) 0.78 + 1.96 * 0.145 = 1.06 , and
b) 0.78 - 1.96 * 0.145 = 0.49

Table 5.4.4-2. Offline analysis results for Patient Number 11.

Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K
P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$
0.89	0.89	0.89	0.49	0.14	1.06 and 0.49

The value of k represents the strength of agreement between the two diagnoses.

5.4.5 SMS - Kappa Analysis for Complete Data

The values in the table below are taken for the offline analysis data sheet of FLMS diagnosis for detail see Appendix C.

Table 5.4.5-1. Kappa Analysis table for complete data.

	Expert (+ve)	Expert(-ve)	Total
SMS (+ve)	39	20	59
SMS (-ve)	1	103	104
Total	40	123	163

Based on the data from Table 5.4.5-1 the Kappa analysis was carried out for the whole patient dataset using the above equations. The overall Kappa value was K = 0.70.

Table 5.4.5-2. Offline Analysis of the whole data.

Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K
P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$
0.87	0.79	0.91	0.57	0.06	0.82 and 0.58

Table 5.4.5-3. K-values Expressed as Strength of Agreement [125].

K – Value	Strength of Agreement beyond chance
<0	Poor
0 - 0.2	Slight
0.21 – 0.40	Fair
0.41 – 0.6	Moderate
0.61 – 0.8	Substantial
0.81 - 1	Almost perfect

For the offline-tests (retrospective analysis) performed, the Kappa based statistical analysis showed substantial level of agreement (k = 0.70) between the experts' and SMS's diagnoses.

The next section gives detailed description of a second type of monitoring system which has been developed using the fuzzy logic for the detection and monitoring of anaesthesia related events.

5.5 Fuzzy Logic Monitoring System (FLMS)

This section gives the detail of FLMS with its overview, design, and working for the better understanding of the system and its operation. Towards the end of this section the results will be discussed.

5.5.1 FLMS Design Overview

Figure 5.5.1-1 shows the building blocks of the FLMS system; it is classified into nine main sections. The top three boxes are explained in detail as the pre-processing part of this project in Chapter 2 and the bottom six boxes are explained in detail in each section of fuzzy logic Chapter 4. For better understanding of this overview, the following points are explained:

1. *Raw Data* – The patient data which contains noise and some artefacts and needs to be filtered, is called the raw data.
2. *Filtering* – The number of different filtering techniques which were applied to reduce the noise and major artefacts to provide the cleaned data for the diagnosis process is also explained in Chapter 2.
3. *Filtered Data* – After the patient data is filtered, this filtered data is given to the FLMS structure for diagnosis.
4. *Clustering* – Two types of clustering techniques were adopted: subtractive and fuzzy-c-means clustering to find the data clusters' centres, and MFs.
5. *FIS* – Fuzzy Inference Structure has to be created for training the patient data through the ANFIS system.
6. *ANFIS Training* – ANFIS structure, which is built on the basic FIS, is used for classification of the patient data.
7. *MFs* – 3 Membership Functions were created for each input: HR-3 MFs, BP -3 MFs, PV -3MFs.
8. *Rules* – The rules are created using all 9 MFs (3x3MFs), mapping the best output for each input.
9. *Warning/Alert* – This is the output where the FLMS gives the warning or an alert for the hypovolaemia in a patient as mild, moderate, or severe.

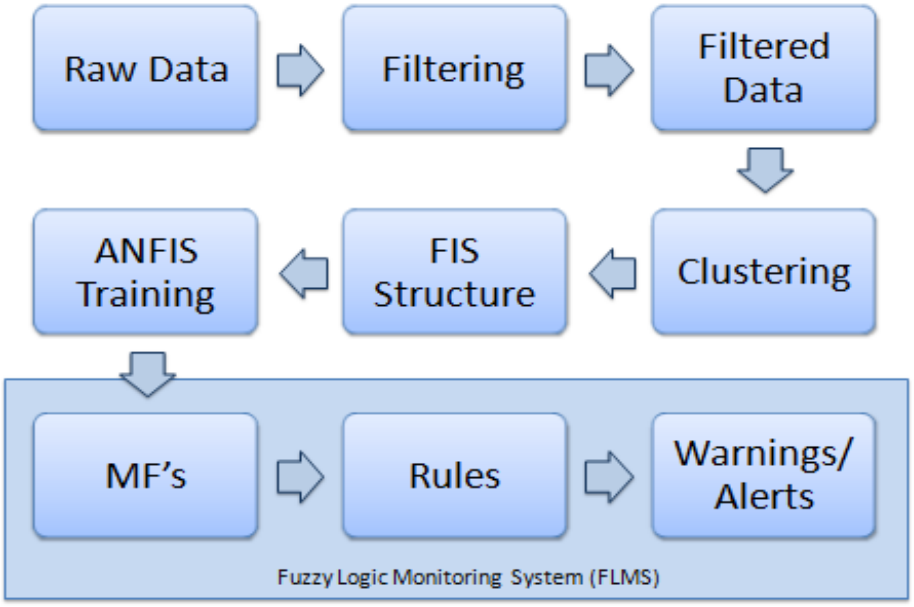


Figure 5.5.1-1. Block diagram of the complete FLMS structure.

5.5.2 FLMS Structure Overview

The FLMS system overview includes all the logic, technique, modules, and major contents.

Figure 5.5.2-1 shows the final structure of FLMS. It was classified into six main sections. Each one of these conditions and modules is explained in detail in Chapter 4 (Fuzzy logic).

1. *ANFIS* – It has been used to train the patient data for the FLMS (Mamdani type model).
2. *FIS* – It is the basic structure which is created for the ANFIS system to build for training and testing the patient's data. This is of Sugeno type model.
3. *MFs* – One of the important parts of the system. The MFs for each input are set as mild, moderate, and severe. The selection of the MF's limits is set after analysing the clustering and ANFIS outputs.
4. *Rules* – These are set using all the MFs and all possible levels of hypovolaemia which were detected throughout the training sessions.
5. *Sugeno Model* – This model is used for ANFIS training and testing.
6. *Mamdani Model* – This model is used as the testing system FLMS.

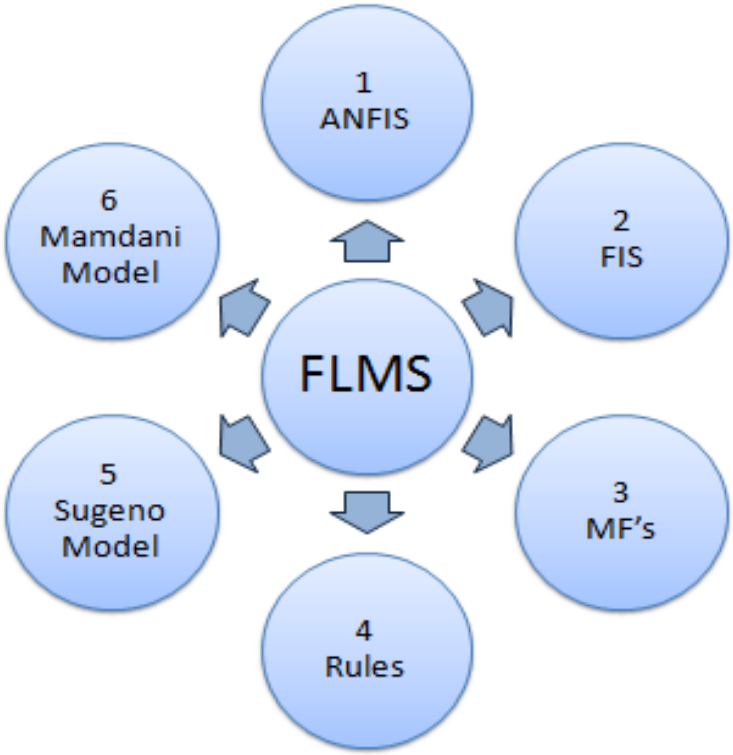


Figure 5.5.2-1. Block diagram of FLMS structure with its 6 major contents, where ANFIS is adaptive neuro- fuzzy inference system, FIS is fuzzy inference system, MFs is membership functions.

5.5.3 FLMS flowchart

This section presents the flow chart of the FLMS structure.

Figure 5.5.3-1 shows the step by step procedure of the FLMS structure. The description of the flow chart diagram is given below.

1. *Raw Data* – Patient data which contains noise and artefacts which need to be pre-processed for better diagnosis and higher performance of the system.
2. *Filtering Techniques* – The raw data was filtered using low pass and variance based techniques.
3. *Batch Process (5 min)* – The total data is divided into five minute intervals for better accuracy and fault detection. The five minutes of data was used as a batch and given to the FLMS for testing; later in some patients the five minute interval was increased to 10 minutes and further to 15 minutes, depending upon the data and its quality.
4. *3 I/P's: Condition 1:* The system checks for the detection of the three inputs. The data given as input should be HR, BP, and PV.

5. *3x3 MFs: Condition 2:* The membership function for each input is set as mild, moderate, and severe. This condition checks the limits and range value of each input MFs. All nine MFs should be mapped to their corresponding input and output spaces for this condition to be true.
6. *7 Rules: Condition 3:* The rules are set for the testing of patient data with their inputs and MFs and their rule weight at the end of each rule. The graphical mapping and the seven rules structure is shown in the Figure 5.6.1-1;
   I. If (ECG-HR is mild) and (BP is mild) and (PV is mild) then (HYPOVOLAEMIA is mild) (1).
   II. If (ECG-HR is moderate) and (BP is moderate) and (PV is moderate) then (HYPOVOLAEMIA is moderate) (1).
   III. If (ECG-HR is severe) and (BP is severe) and (PV is severe) then (HYPOVOLAEMIA is severe) (1).
   IV. If (ECG-HR is mild) and (BP is mild) and (PV is moderate) then (HYPOVOLAEMIA is moderate) (1).
   V. If (ECG-HR is mild) and (BP is moderate) then (HYPOVOLAEMIA is mild) (1).
   VI. If (ECG-HR is mild) and (BP is mild) and (PV is severe) then (HYPOVOLAEMIA is moderate) (1).
   VII. If (ECG-HR is mild) and (BP is severe) and (PV is moderate) then (HYPOVOLAEMIA is moderate) (1).
7. *Warnings/Alarms* – If these three conditions test true, the system generates the output as the warning for Hypovolaemia - mild, moderate, or severe.

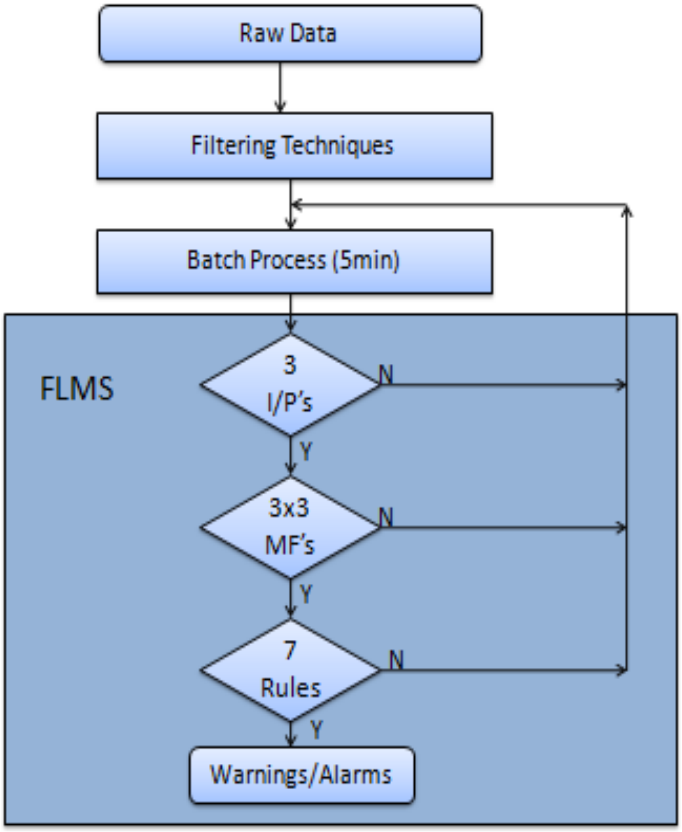


Figure 5.5.3-1. Flow chart of the FLMS system.

5.6 FLMS Testing

5.6.1 FLMS Rules Structure

Figure 5.6.1-1 shows the FLMS structure mapping, shown from left to right:

- Three inputs (black) are mapped to 3x3 input MFs (white).
- Input MFs (white) are mapped to seven rules (blue).
- Seven rules (blue) are mapped to seven output MFs (white) then to one output function (white).
- One output MF (white) is mapped to one output (black) 'Hypovolaemia'.

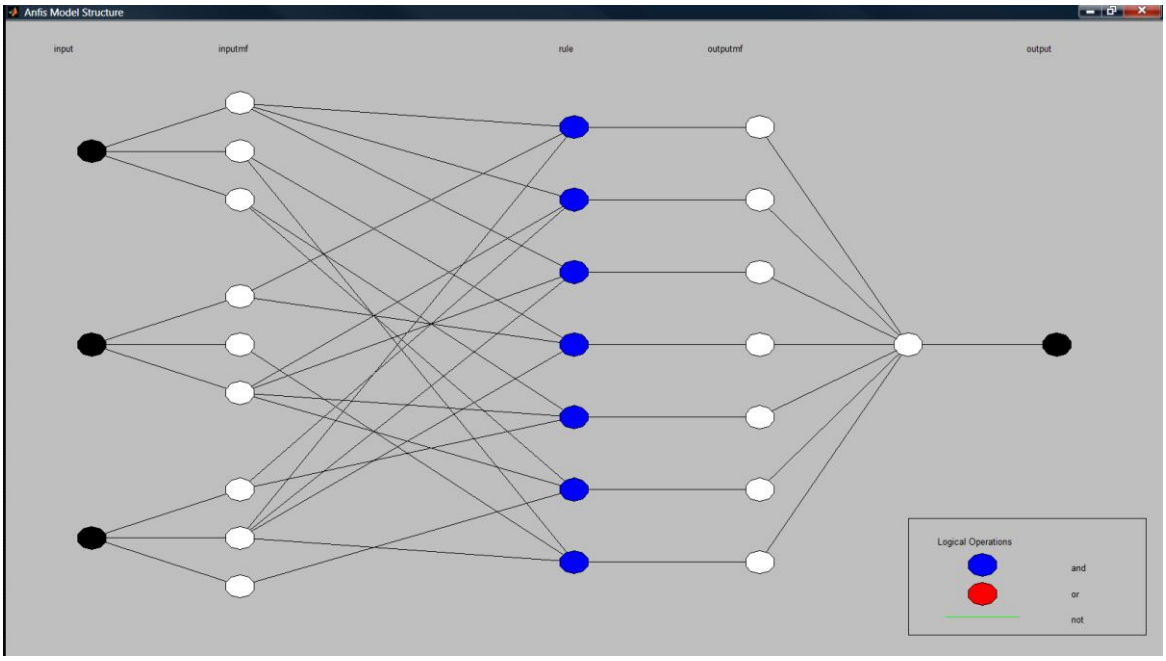


Figure 5.6.1-1. FLMS rules structure.

5.6.2 FLMS's Evaluation and analysis

The objective of the FLMS was to develop an anaesthesia monitoring using fuzzy logic that could help in fault detection in anaesthesia. FLMS is designed as an assistant to the anaesthetist in that it aims to detect problems that can occur during anaesthesia by analysing physiological signals. The advantage of FLMS is that it provides a measure of objectivity and vigilance.

This diagnostic method is completely different to the previously described SMS system. The limits in the FLMS are set to detect the changes in the physiological parameters rather than the crisp numerical value. This approach would give the FLMS more accuracy and a higher degree of diagnosis. Thus, for evaluating FLMS' diagnostic performance, the level of agreement between FLMS and the anaesthetists is used. The following section shows the diagnosis results of FLMS.

5.6.3 FLMS Diagnosis

The patients' data used here has been used in several research projects, and gives the researcher the ability to compare techniques. These data were used for offline data-simulation and analysed retrospectively by revisiting all the available patient information for each patient record. The analysis of this data set in offline mode is identical to the real-time analysis as the same diagnostic algorithms are used for offline analysis as for online analysis.

The physiological data from 15 patients were divided into sub intervals of five minutes, and every three intervals re-combined to a 15 minute time slot. Epochs of 15-minutes duration each were used for offline analysis to match the anaesthetist's diagnosis which is made at 15-minute time intervals.

FLMS's offline version is capable of reading the given patient data files. It analyses and generates the alarm levels for the complete data set. The data from the digital (Excel) patient files was fed to FLMS and the corresponding computer-generated diagnostic alarm levels were produced by the FLMS. The output of FLMS has been classified into three categories: mild, moderate, and severe. These criteria were used and applied to the FLMS for the generation of alerts from the given data sets. The following describes the testing limits and conditions set for testing the FLMS.

5.6.3.1 Condition-1 Three Inputs

Condition-1 is set for the input parameters because, in case of hypovolaemia, the system is set so that it should accept only the three inputs which are HR, BP, and PV. In this condition, if any one data set is missing, the system will return the present data set status as false, and go for the next 15 minutes data set. These three physiological parameters are the main signals for the detection of hypovolaemia.

5.6.3.2 Condition-2 MFs

In Condition-2 of the FLMS system the main MFs were set (including the analysis and limits); the testing of the system was done based on these settings.

5.6.4 Analysis for MFs

The membership function values were set after analysis of a number of important and minor points from the patients' data in those three physiological parameters; those points are:

- The limits of this system are set so that it can detect the changes in the parameters, rather than crisp numerical value as a limit.
- After filtering the data, the patients' data were divided into five-minute time intervals.
- Analysing the average for each parameter (Heart Rate) is done by the division of whole HR data by the average of complete data set.

- = HR / HR AVERAGE (whole set)

- Analysing the five minutes averaging for each patient is done by subtracting the five minute data of HR from the average value of five minute data.
  - = HR (5min) – average value of five minute data
- Analysing the SD changes in the whole data set is done.

These were some of the analyses carried out before defining the limits of MFs of FLMS system.

5.6.4.1 Limits for MFs

Table 5.6.4-1 shows the testing limits set in the Condition-2 of the FLMS. Each MF has been classified into three types - mild, moderate, and severe (Hypovolaemia).

1. Heart Rate –
   1.1. Mild – It has been set from 1.75SD - 3SD.
   1.2. Moderate – It is set from 3SD - 5SD.
   1.3. Severe – It is set from 5SD and above as severe Hypovolaemia.
2. Blood Pressure –
   2.1. Mild – It has been set from 2.75SD - 5SD.
   2.2. Moderate – It is set from 5SD - 6SD.
   2.3. Severe – It is set from 6SD and above as severe Hypovolaemia.
3. Pulse Volume –
   3.1. Mild – It has been set from 4D - 6SD.
   3.2. Moderate – It is set from 6SD - 8SD.
   3.3. Severe – It is set from 8SD and above as severe Hypovolaemia (condition-2).

Table 5.6.4-1. Condition-2 testing SD values and limits for FLMS.

HYPOVOLAEMIA	MILD	MODERATE	SEVERE
Heart Rate (SD)	1.75 – 3	3 – 5	5 & >
Blood Pressure (SD)	2.75 – 5	5 – 6	6 & >
Pulse Volume (SD)	4 – 6	6 – 8	8 & >

5.7 FLMS's Result

Based on the above classification rules and MFs the entire patient's data has been monitored and a comparison chart constructed.

5.7.1 FLMS's Analysis for Patient Number 11

Table 5.7.1-1 shows a sample comparison chart for Patient Number 11. Similar comparison charts were constructed for all 15 patients. Based on the agreement between the expert's and SMS's diagnosis, each of these was classified into three categories: mild, moderate, and severe. Some important points for better understanding the results and their comparison with the anaesthetist's diagnosis are given below:

- Column 1 – Time 15 minutes: It is presented in the table to match the anaesthetist's diagnosis of 15-minute time interval with FLMS.
- Column 2 – Anaesthetist's diagnosis: This diagnosis was provided by Dr. Harrison; it is taken as the comparison with the results of FLMS.
- Column 3 – FLMS Result: In the output section hypovolaemia is classified into mild, moderate, and severe.

The blank spaces in the table are the normal situations where no alert/ warning occurred.

Table 5.7.1-1. SMS diagnosis result (for Patient Number 11).

Patient – 11 Filename – 140408 (In record it's 'Pat11 140408')				
Time 15 minutes	Anaesthetist's diagnosis	Fuzzy Logic Monitoring System (FLMS)		
		MILD	MODERATE	SEVERE
0800				
0815				
0830	N	N	N	N
0845	N	N	N	N
0900	N	N	N	N
0915	N	N	N	N
0930	N	N	N	N
0945	N	P		
1000	N	P		
1015	N	N	N	N
1030	P		P	P
1045	P			P
1100	P			P
1115	P			P
1130	P		P	P
1145	P	P	P	
1200	P	P		
1215	P	N	N	N
1230	N	N	N	N
1245	N	N	N	N
1300				

Where, N = No, P = Possible.

5.7.2 FLMS's Result vs. Anaesthetist's Diagnosis Result

This section explains the result by comparing the result of FLMS with the result of the anaesthetist for Patient Number 11.The Table 5.7.2-1 shows the level of agreement of diagnosis between the FLMS offline results and the expert's comment. Based on the agreement between the expert's diagnosis and FLMS's diagnosis, the complete data set was classified into four categories (TruePOS, TrueNEG, FalsePOS, and FalseNEG). This analysis has been carried out to test the FLMS overall result and performance when compared with other systems.

Table 5.7.2-1. Patient – 11 Filename – 140408 (In record it's 'Pat11 140408')									
Time 15 minutes	Anaesthetist's diagnosis	Fuzzy Logic Monitoring system (FLMS)			Anaesthetist's Comments	Agreement		Disagreement	
		MILD	MODERATE	SEVERE		True POS FLMS +ve Expert +ve	True NEG FLMS -ve Expert -ve	False POS FLMS +ve Expert -ve	False NEG FLMS -ve Expert +ve
0800									
0815					Induction 0824 (intubated 8:28)				
0830	N	N	N	N	Still setting up		1		
0845	N	N	N	N	Still setting up		1		
0900	N	N	N	N	Central line in		1		
0915	N	N	N	N	-		1		
0930	N	N	N	N	Incision		1		
0945	N	P			-			1	
1000	N	P			Dissection CVP 4			1	
1015	N	N	N	N	Dissection CVP 4		1		
1030	P			P	Dissection CVP 4	1			
1045	P			P		1			
1100	P			P		1			
1115	P			P	Morphine given	1			
1130	P			P		1			
1145	P	P	P		Volume loading started 11:50	1			
1200	P	P			Volume loading continuing	1			
1215	P	N	N	N	Volume loading continuing				1
1230	N	N	N	N			1		
1245	N	N	N	N			1		
1300									
TOTAL						7	8	2	1

In the first column (time15 minutes) of the Table 5.7.2-1 shows the three time slots in red colour, that is:

- 09:45 and 10:00 – Shows the expert diagnosis N (no) and FLMS's result shows there is a mild level of Hypovolaemia. From the time slots 09:30 to 09:45 and 09:45 to 10:00 there may be some possibility of Hypovolaemia because the expert's assessment has shown this possible for the next 15-minute slot (10:15 to 10:30). As you can see, there is P (possible) shown by the expert in the next time interval from 10:30 to 10:45. FMS's diagnosis detected mild Hypovolaemia in Patient Number 11 from 09:57 to 10:09 and the warning was generated.
- 12:15 – Expert's diagnosis is P (possible) and FLMS's result shows that there is N (no) Hypovolaemia. From the time slot 12:30 to 12:45, it may not be any possibility of fault because the expert's assessment for the last 30-minute slot of data shows that there is N (no) from 12:30 to 13:00. FMS diagnosis last detected the mild level of Hypovolaemia in Patient Number 11 at around 11:55 and a warning was generated.

These are some issues which need explanation when analysing the FLM system's performance. The next chapter (6) explains the details, and discusses the outcome of FLMS and also compares the results with other expert systems.

5.7.3 FLMS Result Analysis

In this section, Kappa analysis was performed on the results of FLMS system (see 5.4.3 for details of Kappa analysis).

5.7.4 FLMS – Kappa Analysis for Patient Number 11

The values in the kappa analysis are taken from the Table 5.7.2-1.

Table 5.7.4-1. Kappa Analysis table for patient Number 11.

	Expert (+ve)	Expert(-ve)	Total
FLMS (+ve)	7	2	9
FLMS (-ve)	1	8	9
Total	8	10	18

The value for K is:

K = 0.68

Table 5.7.4-2. Offline analysis results for patient #11.

Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K
P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$
0.84	0.82	0.84	0.50	0.17	1.01 and 0.35

5.7.5 FLMS - Kappa Analysis for Complete Data

The values in the table below are taken for the offline analysis data sheet of FLMS diagnosis for detail see Appendix C.

Table 5.7.5-1. Kappa Analysis table for complete data.

	Expert (+ve)	Expert(-ve)	Total
FLMS (+ve)	37	15	52
FLMS (-ve)	3	108	111
Total	40	123	163

Based on the data from Table 5.7.5-1, the Kappa analysis is carried out for the whole patients' dataset.

The K value for overall result was K = 0.73

Table 5.7.5-2. Offline Analysis of whole data.

Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K
P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$
0.89	0.80	0.92	0.59	0.06	0.85 and 0.61

Table 5.7.5-3. K-values Expressed as Strength of Agreement [125].

K – Value	Strength of Agreement beyond chance
<0	Poor
0 - 0.2	Slight
0.21 – 0.40	Fair
0.41 – 0.6	Moderate
0.61 – 0.8	Substantial
0.81 - 1	Almost perfect

Thus for the offline-tests (retrospective analysis) performed, the Kappa-based statistical analysis showed substantial level of agreement (k = 0.70) between the expert's and FLMS's diagnoses.

5.8 Comparison of Results

5.8.1 SMS vs FLMS

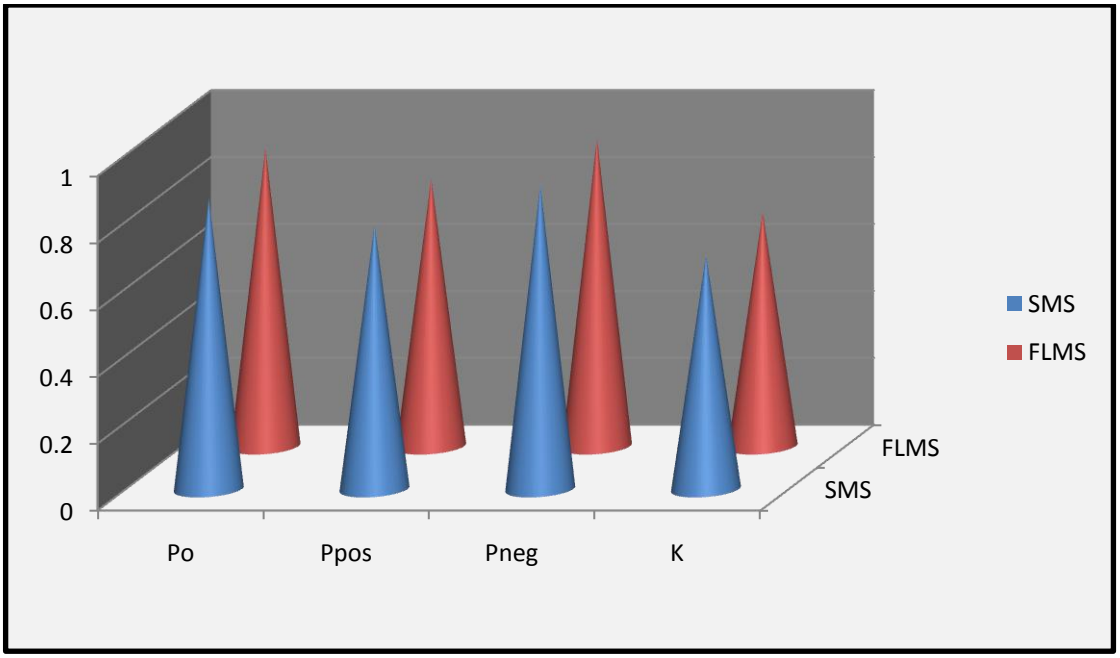
Table 5.8.1-1. Comparing the results of SMS and FLMS.

Monitoring Systems	Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K	Kappa Value
Classification	P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$	K
SMS	0.87	0.79	0.91	0.57	0.06	0.82 and 0.58	0.70
FLMS	0.89	0.80	0.92	0.59	0.06	0.85 and 0.61	0.73

From the table above, the values are taken for the graph 5.1 it shows:

- The comparison of results between two proposed systems (SMS and FLMS).
- X-axis: the values from kappa analysis of two monitoring systems (SMS and FLMS).
- Y-axis: values of each kappa analysis.

Graph 5.1: Comparison of output levels of SMS and FLMS Diagnosis.



5.8.2 Comparing the Results of SMS and FLMS with RT-SAAM [87]

In this section, we compare the overall results of SMS, FLMS, and RT-SAAM.

The Table 5.8.2-1is the offline Kappa analysis results of RT-SAAM published in his work [87]. We compare these results with the results of SMS and FLMS which were shown above.

Table 5.8.2-1. Offline analysis results of RT-SAAM [87].

Overall Aggreement	Positive Agreement	Negative Agreement	Agreement by chance	Standard Error	95% Confidence Intervals for k
P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$
0.81	0.83	0.79	0.50	0.06	0.73 and 0.51

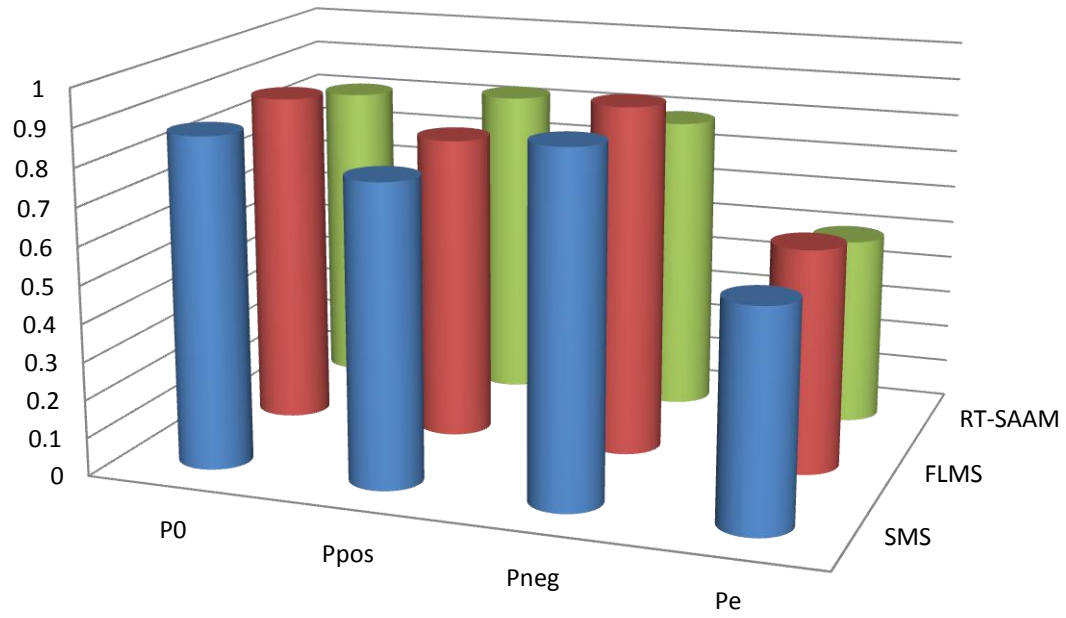
Table 5.8.2-2. Comparing the results of SMS, FLMS, and RT-SAAM.

Monitoring Systems	Overall Agreement	Positive Agreement	Negative Agreement	Agreement by Chance	Standard Error	95% Confidence Intervals for K	Kappa Value
Classification	P_o	P_{pos}	P_{neg}	P_e	SE	$CI_{95\%}$	K
SMS	0.87	0.79	0.91	0.57	0.06	0.82 and 0.58	0.70
FLMS	0.89	0.80	0.92	0.59	0.06	0.85 and 0.61	0.73
RT-SAAM	0.81	0.83	0.79	0.50	0.06	0.73 and 0.51	0.62

5.8.3 Graphical Chart

The chart below compares the P_o, P_{pos}, P_{neg} and P_e values of the three monitoring systems.

Graph 5.2: Comparison of Po, Ppos, Pneg and Pe values of the three monitoring systems.

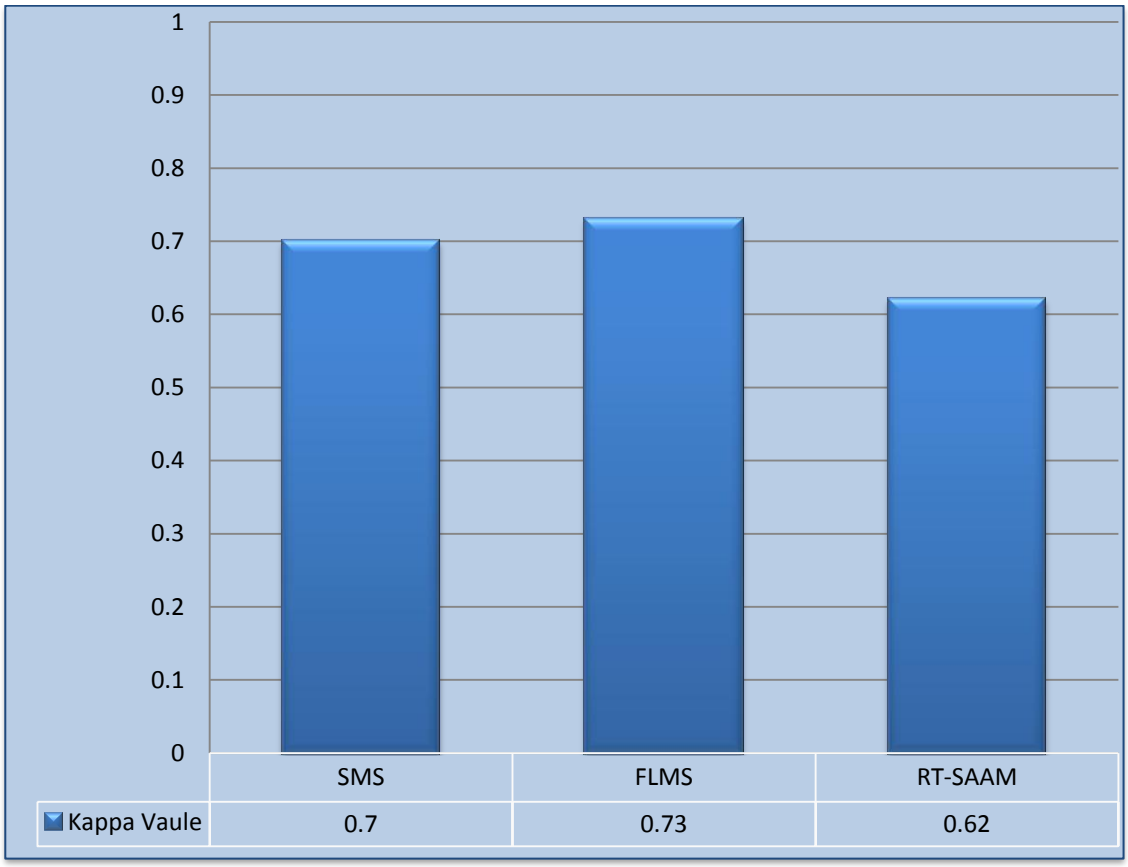


	P0	Ppos	Pneg	Pe
SMS	0.87	0.79	0.91	0.57
FLMS	0.88	0.8	0.92	0.59
RT-SAAM	0.81	0.83	0.79	0.5

5.8.4 Kappa Value Comparison

The chart below compares the Kappa values of SMS, FLMS, and RT-SAAM.

Graph 5.3: Comparison of Kappa Analysis of SMS, FLMS and RT-SAAM.



5.9 Summary

The SMS and FLMS were both successfully tested in offline mode. The initial result shows the higher efficiency and accuracy in both monitoring systems with the K value of 0.70 in both the systems when compared with 0.62 of RT-SAAM. All of the above graphs show very close results between the computer systems (SMS and FLMS) and the anaesthetist's diagnosis. The complete results of all the patients are then compared. The final performance results of SMS and FLMS place the system performance at a substantial level.

The SCADA monitoring system has its own unique characteristics, and it is a fact that SCADA systems are the most popular industrial monitoring systems in many industrial fields, with their superior quality and user-friendliness for controlling plants and

monitoring processes. The working of SMS is explained in detail above with its complete overview, design, structure, conditions, and diagnosis environment.

Fuzzy Logic is a very widely used technique in many applications today. It can be used in complex situations with great accuracy. The overviews, FLMS design and structure, limits, rules, and MFs are explained in the above sections.

The next chapter concludes the thesis with a discussion of these systems and future work.

CHAPTER 6 Discussion and Conclusions

6.1 Introduction

The use of monitoring systems to automate medical diagnoses may improve the quality of care. Two monitoring systems have been described that could be used during anaesthesia to monitor some of the most important physiological parameters. These monitoring systems have been validated using one particular condition called hypovolaemia. For each parameter, three simulated scenarios including mild, moderate, and severe levels of the output have been considered to assess the performance of the monitoring systems.

6.1.1 Major Learning Outcomes

Some of the important areas covered in this project work are:

- Data conversion: The process of data conversion starts with data collection from the patient at the time of the operation and continues until this information is saved in the text file. For more details, refer to Chapter 2.
- Medical Information: Complete understanding of the medical side of this project work was carried out with the help of an anaesthetist, who showed, how these signals are generated by the patients, and which are the most important signals for hypovolaemia, operating theatre setup, and the equipment used by anaesthetists during operations. This can be found in the first chapter.
- SCADA Software: The use and working of this software in the area of clinical monitoring was studied before implementing it in this project See Chapter 3.
- Fuzzy Logic: Fuzzy logic has been studied for the development of the FLMS system; the important areas in fuzzy logic that were investigated are ANFIS, FIS, clustering, and different types of fuzzy models. Details can be found in Chapter 4.

A detailed discussion is presented on some of the important stages of this research project and the conclusions section describes the research outcome and addresses further research questions. The future work section shows the work for further refining of the system and improving the performance for other possible applications.

6.2 Discussions

The performance of the SMS and FLMS systems for assessment of hypovolaemia varies with input features selected. To extract usable features from the signals, further processing needed to be done on the signal including artefact rejection from the raw physiological data. The fine tuning of raw data was one of the main building blocks of this project. To this end, a number of different filtering techniques were used such as low pass, high pass, and variance-based filtering. The variance-based filtering technique (discussed in Chapter 3) was used for eliminating major artefacts. The variance-based filtering algorithm analyses variance of data-batches, which represent a segment of the whole time series data, and filters the time-series data on a batch-by-batch basis. The limits for the variance-based filters were generalized during the data-analysis stage using the whole patient data.

6.2.1 SMS

6.2.1.1 SMS Development

The problem with simple threshold alarms is that up to 94.5percent of the alarms that sound in the ICU are false, are provider-induced, [59] and frequently sound unnecessarily [56, 57, 59]. Default settings by the equipment manufacturers are set to avoid missing a single false negative alarm, and thereby result in many false positive alarms [60].

The use of SCADA software has been demonstrated in several applications and studies. It has been shown that this system can be used in the area of patient monitoring to provide a useful clinical assessment. The aim was to develop a trustworthy tool for managing safe and balanced anaesthesia monitoring. SMS was designed and built for this research work. The clinical results indicate that SMS is a promising tool in the anaesthesia monitoring field, when compared with other systems such as:

Charbonnieran et al. [72] developed an on-line segmentation algorithm to pre-process data describing the patient's state. Threshold alarms were evaluated based on original observations (classified by expert as true, false, and technical alarm). All 19 true alarms and six technical alarms were detected. Twenty out of the 24 false alarms were rejected. No extra alarms were generated.

Gohil et al. [32] developed the diagnostic system which is called Real-Time – Smart Alarms for Anaesthesia Monitoring (RT-SAAM). Two diagnostic modules were developed: a probabilistic alarms (Probabilistic Module) and respiration-induced systolic pressure variations (SPV Module). The accuracy of the diagnostic results from RT-SAAM was analyzed using Kappa analysis. The results showed that RT-SAAM achieved 75 percent overall agreement, and is capable of diagnosing the pathological events with a substantial to fair level of agreement between RT-SAAM and the anaesthetist.

Ansermino et al. [33] developed a software tool (iAssist) which tracks the statistical properties of multiple dynamic physiological processes and identifies new trend patterns. The Cumulative Sum (CUSUM), technique is used and as a result the software tool (iAssist) correctly identified 90 percent (869/960) changes at a rate of 12.9 per hour.

SMS system in its offline testing achieved the overall agreement of 87 percent and is capable of diagnosing with a substantial level of agreement between SMS and the anaesthetist. This is because of the combination of crisp values limits (condition 1) and the SD changes (condition 2) of each parameter used as the main conditions for SMS diagnosing.

6.2.1.2 SMS Advantages

The SMS has the following advantages:

1) Monitoring: records the anaesthesia data in real-time using a graphical display.
2) Supervisory: records the anaesthesia data in real-time with changing set points using the serial port for communication. Stores the data from all devices and user inputs in a file with alarms and historical data.
3) Alarm: generates alarms for events that may occur during the anaesthesia, with a graphical signal or/and dialogue box in the alarm chart so as to warn the anaesthetist.
4) Control: has the ability to incorporate mathematical algorithms and logical models, in order to detect the level of hypovolaemia.
5) Data Stored: any data provided by the alarm chart and historical trend generate a file which is stored with the number of the processed patient.

6) Report: these files are saved in a standard extension such as .doc or.xls at the user's option.

6.2.1.3 Challenges

At the time of the SCADA software installation there was a major installation error due to the use of shared machines at the University. SCADA software cannot be installed in the general shared machines where there will be access by different users without any restriction or security. To overcome this installation process a new virtual machine was installed within the main machine. It works the same as the main machine, but in this virtual machine there will be total control of user's access as the SCADA security installation requirement.

This software works as the master user, which hosts the main service and allows other users entry on secure access with different levels of safety. Other clients can access the project and its digital saved files and information for monitoring from a remote place or far from the main setup. This can be done via the Internet, Ethernet, Local Access Network (LAN) or Wireless Access Network (WAN).

Different versions of SCADA software have been tried. The earlier InTouch version (v7) was tried, but failed as it was not compatible with the Windows XT operating system. InTouch version 9.5 was found to be the most suitable version because this version had better compatibility with the operating system (which our machine had), external links, and data files, and was installed with an upgrade of the Windows service pack.

The second challenge was to learn the SCADA language. This was done by learning the important tools that were appropriate for this project work. It was not possible to learn the whole set of software in the allotted time. Some of the important features learned were: alarm tools, historical graphical trend, graphical displays, tags, animation links, and front panel GUI display.

6.2.1.4 SMS Performance

The SMS anaesthesia monitoring system performed well in diagnosing hypovolaemia during anaesthesia. The system was robust for monitoring within the given limits and conditions. The SMS was tested with physiological data from 15 patients in offline

mode only. The physiological data were divided into 15-minute segments and the diagnoses for each of these were evaluated. The useful diagnoses from the anaesthetist and the SMS were compared to measure the level of agreement.

During the tests (offline), substantial levels of agreement were achieved. In some 15-minute segments, SMS made an early detection in the diagnosis of hypovolaemia, before the anaesthetist's 15-minute time slot review. In some time slots, SMS generated false positive alarms. These time slots were again reviewed to check the false positive alarms. It was found that the limits was not generalised enough. In some patients' data the normal limit was slightly higher than normal; following these alarms, the limits were again refined, resulting in the decrease in the false positive alarms.

Kappa analysis was used for computing the level of agreement between the two results (anaesthetists and SMS) and found the K value for Patient Number 11 to be 0.76. When SMS was tested with complete data sets, the K value was to be 0.70 with overall agreement of 0.87. These results show the system's accuracy and sensitivity. Kappa analysis used in this research is simply to measure the level of agreement between the two observers. This is important as this research aims to equal or improve on anaesthetists' performance; this is a statistics-based analysis that does not conclude that either of the observers is correct. The kappa analysis provides incomplete but very reliable indication of the performance. It is likely that SMS detects the fault in diagnosis when the clinician may not. In this case, SMS could be correct and yet there is disagreement. This shows that even though Kappa statistics can provide a measure of agreement between the two observers, it does not completely represent the actual performance of SMS.

6.2.2 FLMS

6.2.2.1 FLMS Development

An optimally designed alarm system should give the user adequate warning to take appropriate action before a critical situation occurs. On the other hand, the system should avoid generating false alarms. To fulfil these criteria an alarm system needs to detect trends as well as monitor thresholds in order to alert the anaesthetist before the variable reaches a critical level. Conventional limit alarms do not meet these criteria because of their poor performance as trend detectors which cause high false positive rates.

Several trend detecting alarm systems have been designed in such a way that if the adverse condition persists, the alarm will be deactivated. However, for a simpler analysis and display, it is more helpful if just a single variable or alarm condition is given for each parameter being monitored. Some of the systems which are compared with FLMS are listed below:

Becker et al. [8] developed a fuzzy logic intelligent alarm system for online validation of anaesthesia in five hemodynamic state variables. The validation by fuzzy logic system and anesthesiologists were repeatedly compared. Online validation achieved the sensitivity of 99.3 percent and specificity of 66 percent.

Wolf et al. [126] developed a fuzzy logic alarm system with the events classified as movement artefacts, moderate apnea, severe apnea, and hypoxia. As a result, the fuzzy system correctly classified 99.4 percent of all alarms into their respective categories.

Oberli et al. [127] developed a fuzzy logic system with expert observers classifying alarms: True Positive: “correct” or “correct but unnecessary” alarm. The standard monitor achieved sensitivity of 79 percent and the fuzzy logic system achieved sensitivity of 92 percent.

FLMS system in its offline testing achieved the overall agreement of 88 percent and is capable of diagnosing with a substantial level of agreement between FLMS and the anaesthetist. This is because the FLMS has been developed using the MFs and rules to diagnose hypovolaemia. The FLMS system is designed in such a way that it only detects the changes in the physiological parameters with respect to their SD limits rather than normal and abnormal limits. Due to this detection of changes, the FLMS has been able to detect the minute details in some of the 15-minute data sets.

6.2.2.2 FLMS Advantages

The proposed FLMS system has the following advantages:

1) Classification results are improved by comparison with other methods using single features. It can completely discriminate between mild and severe hypovolaemia, and the system can also classify the moderate state.
2) Continuous monitoring of changes from normal level to severe level of hypovolaemia which is easy for the anaesthesiologist to understand.

3) It is easy to fuse and extract knowledge to and from the system as it was initialized by an expert's comments with just seven rules.
4) Independence of the test subject.

6.2.2.3 FLMS Performance

The usefulness of FLMS as an aid to situation assessment is more difficult to determine, particularly in view of the fact that no on-line trials have been undertaken. The results shown in this section are collated from more than 50 hours of data collected by anaesthetists during general surgery over a long period of time.

Furthermore, the justified events were also categorised according to their diagnoses. This was used to test the accuracy of FLMS in correctly identifying problems. The FLMS system is designed for the detection of hypovolaemia with three features and three different levels of output. It works on the changes in parameter as it was found that this system detects the minute details of fault in some of the cases.

Kappa analysis was used for computing the level of agreement between the two results (anaesthetists and FLMS) and found the K value for Patient Number 11 to be 0.66. When FLMS was tested with complete data sets, the K value was found to be 0.70 with overall agreement of 0.88. These results show the FLMS has a substantial level of diagnosis performance.

6.3 Conclusion

The monitoring systems tested in this investigation are not designed to replace clinicians, but rather to assist the anaesthetist in rapidly processing the vast amount of information available from monitoring equipment, and to convey this information in a meaningful manner so that rapid intervention can occur. Usability results indicate that anaesthetists consider this system to be a useful tool in the identification of changes in patient status.

This research has achieved approximately 90 percent of the stated goals. This research has developed two clinically useful diagnostic alarms for anaesthesia. The first thing which has to be remarked is that introducing the SCADA software gave us very good motivation and enthusiasm towards the research work. Hence the conclusion that the experimental and theoretical methods and methodologies used in building our

prototype SMS system were correctly chosen and used. As for positive aspects, it has to be recorded that the capability of the system to diagnose hypovolaemia accurately in such a way that the result can very clearly coincide with the anesthetist's diagnosis has been a worthwhile development.

Furthermore, FLMS proved the capacity of the system to be used in monitoring the changes in each parameter, and more accurately detect differences between the levels of hypovolaemia. Even in the current state, the proposed FLMS system can be used for anaesthesia monitoring and detection of hypovolaemia, which makes it easily utilizable in real-time environments.

The complete validation of both systems as a clinically useful diagnostic alarm system can only be verified after real-time testing. These systems are ready to be tested in the real-time environment, although it is likely they will need further refinement and enhancement with additional features for routine clinical use.

6.4 Future Work

There is room for improvement in the signal processing of the data.

The SMS can be updated for higher performance of the system. Apart from the existing SMS system, the possible important developments are listed below and marked in Figure 6.2.2-1.

1. Sound – Voice prompting is one of the important features of any monitoring system, with the generation of warning/alert that tells the anaesthetists in a clear voice the nature of the alarm state. Anaesthetists are not always in a position to look at the monitor screen.
2. Yes/No – the addition of a yes/no button after the warning gives the anaesthetist the facility to acknowledge the alarm and the choice to reject or accept it.
3. Message – this message window space provides the anaesthetist a place to write comments/suggestions, so that SMS should save this as part of diagnosis rule which will help in the assessment of performance of the SMS system.
4. Shortcuts – this box gives the shortcut to save the data into the drive, save the digital values in the Excel workbook, and print the whole/part of the report.

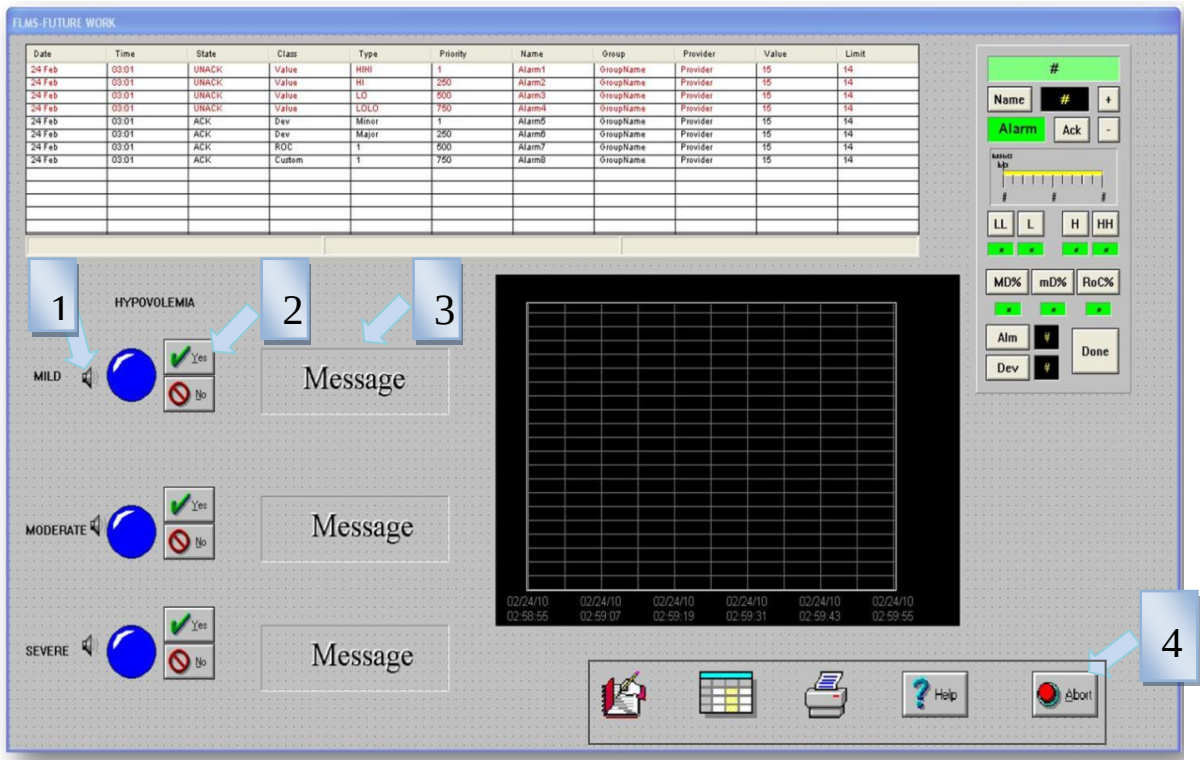


Figure 6.2.2-1. SMS - An outline on the future development.

The FLMS can be improved further in the area of membership functions and rules, as these are the main building blocks for a fuzzy system. There will be room to improve the limits as they differ with each patient. The system, with more generalised rules, may prove more efficient.

To extract more robust features, further work must be done on the signal artefact rejection and de-noising of the raw physiological features. It is intended to add some other features and examine different combinations of them as an input to the fuzzy system. Although these inputs have sufficient information for hypovolaemia, they cannot monitor the complexity of the whole situation. So combining these features with hemodynamic information and also measurements of some other important parameters may result in more confident indices of the performance of FLMS.

This system has been proved to be clinically useful on its own by the overall results, and when compared with other monitoring systems. This confirms the effectiveness of fuzzy logic in patient monitoring, but it still needs to be tested in a real-time environment to show its full clinical worth.

References

[1] D. M. Gaba, K. J. Fish, and S. K. Howard, *Crisis management in anesthesiology*. Churchill Livingstone: New York, 1994.

[2] J. M. Ansermino, "Patient safety: technology and design can help," *B C Medical Journal*, vol. 48, pp. 339–341, 2006.

[3] G. A. Miller, "The magical number seven, plus or minus two: some limits on our capacity for processing information," *Psychological Review*, vol. 101, pp. 343–352, 1956.

[4] S. A. Norman, S. W. Karen, N. A. Grenvik, R. Deborah, and V. S. James, "Adverse Occurrences in Intensive Care Units," *JAMA*, vol. 244, pp. 1582-1584, 1980.

[5] L. G. Deneault, C. M. Lewis, A. Debons, K. L. Stein, and A. M. Dewolf, "An integrative display for patient monitoring," in *Proceedings of the IEEE International Conference in Systems, Man and Cybernetics*, Los Angeles, CA pp. 515-517, 1990.

[6] G. C. Van den Eijkel and E. Backer, "Knowledge acquisition using a fuzzy machine-learning algorithm for a knowledge-based anesthesia monitor," in *18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Amsterdam, pp. 1997-1998, 1996.

[7] J. B. Cooper, R. S. Newbower, C. D. Long, and B. McPeek, "Preventable anesthesia mishaps: A study of human factors," *Quality & Safety in Health Care*, vol. 11, pp. 277-282, 2002.

[8] K. Becker, H. Kasmacher, G. Rau, G. Kalff, and H. J. Zimmermann, "A fuzzy logic approach to intelligent alarms in cardioanesthesia," in *IEEE World Congress on Computational Intelligence*, Orlando, FL pp. 2072-2076, 1994.

[9] I. Nevo, A. Guez, F. Ahmed, and J. V. Roth, "System theoretic approach to medical diagnosis," in *Proceedings of the Fourth Annual IEEE Symposium in Computer-Based Medical Systems*, pp. 94-96, 1991.

[10] "Inner body: Your guide to human anatomy online," 2009.

[11] N. L. O. Medicine, "Anatomy," in *National Institutes of Health* USA, 2009.

[12] G. P. Barash, "Clinical Anaesthesia," in *Clinical Anaesthesia* USA: Lippincott Williams and Wilkins, pp. 210-245, 2008.

[13] M. S. Bhende, "End-tidal carbon dioxide monitoring in pediatrics: concepts and technology." vol. 47, pp. 153-156, 2001.

[14] S. M. Bradley, J. M. Simsic, M. Andrew, and J. L. Myers, "Hemodynamic effects of inspired carbon dioxide after the Norwood procedure," in *The Annals of thoracic surgery*. vol. 72, pp. 2088-2094, 2001.

[15] P. Szolovits and G. P. Stephen, "Computers and Clinical Decision Making: Whether, How, and For Whom?," *IEEE*, vol. 1 pp. 1224-1226, 1979.

[16] E. H. Shortliffe, B. G. Buchanan, and E. A. Feigenbaum, "Knowledge engineering for medical decision making: A review of computer-based clinical decision aids," *Proceedings of the IEEE*, vol. 67, pp. 1207-1224, 1979.

[17] D. Anthony, E. Hines, D. Taylor, and J. Barham, "An investigation into the use of neural networks for an expert system in nuclear medicine image analysis," in *Third International Conference in Image Processing and its Applications*, pp. 338-342, 1989.

[18] E. Czogala, J. Chmielniak, A. Brodziak, and W. Siler, "Decision making by means of probabilistic sets in medical expert systems," in *Proceedings of the Annual International Conference of the IEEE in Engineering in Medicine and Biology Society*, pp. 1378-1379, 1988.

[19] D. A. Linkens and M. F. Abbod, "Intelligent control of anaesthesia," in *Intelligent Methods in Healthcare and Medical Applications (Digest No. 1998/514), IEE Colloquium*, York pp. 2/1-2/4, 1998.

[20] S. Mahalingam, "CSRI-an expert system tool and its applications," in *Proceedings of the IEEE 1989 National Aerospace and Electronics Conference*, pp. 1378-1382, 1989.

[21] S. M. Mohiddin, "Medical differential diagnostic system (DDS) using fuzzy inferencing methods," in *Artificial Intelligence in Medical Decision Making, IEE Colloquium*, pp. 4/1-4/5, 1990.

[22] J. Xue and M. Krajnak, "Fuzzy Expert Systems For Sequential Pattern Recognition For Patient Status Monitoring in Operating Room," in *28th Annual International Conference of the IEEE in Engineering in Medicine and Biology Society*, pp. 4671-4674, 2006.

[23] G. C. van den Eijkel, van der Lubbe, C.A., and E. Backer, "Fuzzy Incremental Learning of Expert Rules for a Knowledge-Based Anaesthesia Monitor.," in *Fourth European Congress on Intelligent Techniques and Soft Computing*, Germany, 1996.

[24] E. Coiera, "Clinical decision support systems," in *A Guide to Health Informatics*, 2 ed, 2003.

[25] A. H. Morris, "Developing and implementing computerized protocols for standardization of clinical decisions," *Annals of Internal Medicine*, vol. 132, pp. 373-383, 2000.

[26] R. C. Watt, E. S. Maslana, and K. C. Mylrea, "Alarms and anesthesia: challenges in design of intelligent systems for patient monitoring," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 12, pp. 34-41, 1993.

[27] D. Dunsmuir, J. Daniels, C. Brouse, S. Ford, and J. Ansermino, "A knowledge authoring tool for clinical decision support," *Journal of Clinical Monitoring and Computing*, pp. 189-198, 2008.

[28] D. Mouloud, M. Mahdi, and R. Jonathan, "A Fuzzy Decision Support System for Therapy Administration in Cardiovascular Intensive Care Patients," *IEEE International Fuzzy Systems*, pp. 1-6, 2007.

[29] J. B. Fitzmaurice, P. K. Nicholas, M. Martin, S. Chase, P. Ford-Carleton, G. Estey, R. Zielstorff, and O. Barnett, "Evaluation of a problem-based access knowledge system using triangulation methods," *Medinfo*, vol. 8 pp. 1059-1063, 1995.

[30] J. Liu, J. Wyatt, and D. Altman, "Decision tools in health care: focus on the problem, not the solution," *BMC Medical Informatics and Decesion Making*, p. 4, 2006.

[31] E. Coiera, "Intelligent monitoring and control of dynamic physiological systems," *Artificial Intelligence in Medicine,* vol. 5, pp. 1-8, 1993.

[32] B. Gohil, H. Gholam Hosseini, M. Harrison, A. Lowe, and A. Al-Jumaily, "Intelligent Monitoring of Critical Pathological Events during Anesthesia," in *Engineering in Medicine and Biology Society Conference*. vol. 29 France: IEEE, pp. 4343-4346, 2007.

[33] J. M. Ansermino, P. D. Jeremy, T. Randy, L. Joanne, Y. Ping, J. B. Chris, A. D. Guy, and B. John, "An Evaluation of a Novel Software Tool for Detecting Changes in Physiological Monitoring," *International Anesthesia Research Society, ANESTHESIA & ANALGESIA,* vol. 108, pp. 873-880, 2009.

[34] P. Gunther, F. Kimiko, H. Volker, M. Peter, V. Andreas, B. Luzius, K. Andrea, F. Yoshihisa, I. Daniel, and L. Daniel, "Automatic Algorithm for Monitoring Systolic Pressure Variation and Difference in Pulse Pressure," *International Anesthesia Research Society, ANESTHESIA & ANALGESIA,* vol. 108, pp. 1823 - 1829, 2009.

[35] P. Grant and O. Naesh, "Fuzzy logic and decision-making in anaesthetics," *Journal of the Royal Society of Medicine,* vol. 98, pp. 7-9, 2005.

[36] T. Sieber, C. Frei, M. Derighetti, P. Feigenwinter, D. Leibundgut, and A. Zbinden, "Model-based automatic feedback control versus human control of end-tidal isoflurane concentration using low-flow anaesthesia," *British Journal of Anaesthesia,* pp. 818-25, 2000.

[37] A. Carregal, A. Figueira, and M. Nunez, "Fuzzy logic and postoperativepain," *Rev Esp Anestesiol Reanim* vol. 44, pp. 215-17, 1997.

[38] A. Lowe, "Evidential Inference for Fault Diagnosis," in *Engineering* Auckland: University of Auckland, 1998, p. 217.

[39] A. Lowe and M. J. Harrison, "Computer-enhanced diagnosis of malignant hyperpyrexia," *Anaesthesia and Intensive Care [NLM - MEDLINE]*, vol. 27, p. 41, 1999.

[40] V. Esmaeilia, A. Assarehb, Shamsollahia, M. H. Moradib, and N. M. Arefianc, "Estimating the depth of anesthesia using fuzzy soft computation applied to EEG features," *Intelligent Data Analysis,* pp. 393-407, 2008.

[41] C. S. Nunesa, M. Mahfouf, and D. A. Linkensb, "Fuzzy modelling for controlled anaesthesia in hospital operating theatres," *Control Engineering Practice,* pp. 563-572, 2006.

[42] M.J. Harrison, M.T. Kluger, and N.N. Robertson, "The relationship between change in blood pressure, blood pressure and time," *Anaesthesia* vol. 55, pp. 385-387, 2000.

[43] M. Harrison and C. Connor, "Statistics-based alarms from sequential physiological measurements," *Anaesthesia,* pp. 1015-1023, 2007.

[44] M. Dosani, J. Lim, P. Yang, C. Brouse, J. Daniels, G. Dumont, and J. M. Ansermino, "Clinical evaluation of algorithms for context-sensitive physiological monitoring in children," *British Journal of Anaesthesia,* vol. 102, pp. 686-91, 2009.

[45] T. W. Schnider, C. F. Minto, P. L. Gambus, C. Andresen, D. B. Goodale, S. L. Shafer, and E. L. Youngs, "The Influence of Method of Administration and

Covariates on the Pharmacokinetics of Propofol in Adult Volunteers," *Anesthesiology,* vol. 88, pp. 1170-82, 1998.

[46] C. F. Minto, T. W. Schnider, and S. L. Shafer, "Pharmacokinetics and pharmacodynamics of remifentanil. II. Model application, ," *Anesthesiology,* vol. 86, pp. 24-33, 1997.

[47] N. Bressan, A. Castro, B. Susana, H. Oliveira, L. Ribeiro, A. David, P. Amorim, and S. Catarina, "Synchronization Software for Automation in Anesthesia," in *29th Annual International Conference of the IEEE EMBS*, Lyon, France, 2007.

[48] P. Schreiber and J. Schreiber, "Structured alarm systems for the operating room," *Journal of Clinical Monitoring* pp. 201-204, 1989.

[49] L. D. James, L. T. Christopher, and G. S. Timothy, "The application of a modified proportional-derivative control algorithm to arterial pressure alarms in anesthesiology," *Journal of ClinicalMonitoring and Computing,* pp. 41-47, 1998.

[50] L. Feng, S. Kin, L. C.M., D. J.M., and K. H. Chon, "A Robust Method for Detection of Linear and Nonlinear Interactions: Application to Renal Blood Flow Dynamics " *Annals of Biomedical Engineering,* vol. 34, pp. 339-353, 2006.

[51] D. D. Woods, R. I. Cook, and C. E. Billings, "The impact of technology on physician cognition and performance," *Journal of Clinical Monitoring,* vol. 11, pp. 5-8, 1995.

[52] R. C. Watt, E. S. Maslana, and K. C. Mylrea, "Alarms and anesthesia: Challenges in the design of intelligent systems for patient monitoring," *IEEE Engineering in Medicine and Biology,* vol. 12, pp. 34-41, 1993.

[53] M. J. Navabi, R. C. Watt, S. R. Hameroff, and K. C. Mylerea, "Integrated monitoring can detect critical events and improve alarm accuracy," *Journal of Clinical Engineering,* vol. 1, pp. 295-306, 1991.

[54] A. K. Ream, *Future trends in monitoring and biomedical instrumentation, In:Monitoring in Anesthesia*, 2 ed. Boston: Butterworth Publishers, 1984.

[55] P. J. Schreiber and J. Schreiber, "Structured alarm systems for the operating room," *Journal of Clinical Monitoring,* vol. 5, pp. 201-204, 1989

[56] M. Imhoff and S. Kuhls, "Alarm Algorithms in Critical Care Monitoring," *International Anesthesia Research Society,* pp. 1525–1537, 2006.

[57] M. Chambrin, P. Ravaux, D. Calvelo-Aros, A. Jaborska, C. Chopin, and B. Boniface, "Multicentric study of monitoring alarms in the adult intensive care unit (ICU): a descriptive analysis," *Intensive Care Med* pp. 1360-6, 1999.

[58] C. Tsien and J. Fackler, "Poor prognosis for existing monitors in the intensive care unit," *Critical Care Medicine,* pp. 614-19, 1997.

[59] S. Lawless, "Crying wolf: false alarms in a pediatric intensive care unit," *Crit Care Med* pp. 981-5, 1994.

[60] M. Quinn, "Semipractical alarms: a parable," *Journal of Clinical Monitoring,* pp. 196-200, 1989.

[61] T. Schecke, G. Rau, H. Popp, H. Kasmacher, G. Kalff, and H. Zimmermann, "A knowledge-based approach to intelligent alarms in anesthesia," *Engineering in Medicine and Biology,* pp. 38-43, 1991.

[62] G. Matthias, A. M. Boaz, and R. W. Dwayne, "Improving Alarm Performance in the Medical Intensive Care Unit Using Delays and Clinical Context," *International Anesthesia Research Society, Anaesthesia & Analgesia,* vol. 108, pp. 1546-52, 2009.

[63] A. T. Rheineck-Leyssius and C. J. Kalkman, "Influence of pulse oximeter settings on the frequency of alarms and detection of hypoxemia: theoretical effects of artifact rejection, alarm delay, averaging, median filtering or a lower setting of the alarm limit," *Journal of Clinical Monitoring and Computing,* pp. 151-156, 1998.

[64] W. Zong, G. B. Moody, and R. G. Mark, "Reduction of false arterial blood pressure alarms using signal quality assessment and relationships between the electrocardiogram and arterial blood pressure " *Medical and Biological Engineering and Computing,* vol. 42, 2006.

[65] W. H. young, R. M. Gardner, T. D. East, and K. Turner, "computerized ventilator data selection: arrifact sejection and data reduction," *journal of Clinical Monitoring and Computing,* pp. 165-171, 1997.

[66] A. Makivirta, E. Koski, and T. sukuvaara, "The median filter as processor for a patient monitor limit alarm system in intensive care," *Computer Methods Programs Biomedical,* pp. 139-144, 1994.

[67] R. Schoenberg, D. Sands, and C. Safran, "Making ICU alarms meaningful: a comparison of traditional vs. trend-based algorithms," *Proceedings of the American Medical Informatics Association Symposium,* pp. 379-383, 1999.

[68] I. M. Trimble, M. West, M. S. Knapp, R. Pownall, and A. F. Smith, "Detection of renal allograft rejection by computer," *British Medical Journal,* pp. 1695-1699, 1983.

[69] C. Cao, N. McIntosh, I. Kohane, and K. Wang, "Artifact Detection in the PO2 and PCO2 Time Series Monitoring Data from Preterm Infants," *Journal of Clinical Monitoring and Computing,* vol. 15, pp. 369-378, 1999.

[70] I. Haimowitz and I. Kohane, "Managing temporal worlds for medical trend diagnosis " *Artificial Intelligence in Medicine,* pp. 299-321, 1998.

[71] E. Koski, T. Sukuvaara, A. Makivirta, and A. kari, "A knowledge-based alarm system for monitoring cardiac operated patients-assessment of clinical performance," *International Journal of Clinical Monitoring and Computing,* vol. 11, pp. 79-83, 1994.

[72] S. Charbonnier, G. Becq, and L. Biot, "On-line segmentation algorithm for continuously monitored data in intensive care units," *IEEE Transactions on Information Technology in Biomedicine,* vol. 51, pp. 484–92, 2004.

[73] M. Harrison, "Accentuating the (true) positives and eliminating the (false) negatives in intra-operative real time diagnostics and decision-making," Auckland, New Zealand, 2003.

[74] A. L. MJ Harrison, RW Jones, "Assessment of heart function using fuzzy templates and evidence based reasoning to process data during anaesthesia," in *ICSC Symposia On Engineering of Intelligent Systems,* 2000.

[75] B. Gohil, M. J. Harrison, H. Gholam Hosseini, A. Lowe, and A. Al-Jumaily, "Computer generated pathophysiological diagnoses during anaesthesia," in *Euroanaesthesia* Munich, Germany, pp. 21, 2007.

[76] H. C. F. M. Association, "Achieving operating room efficiency through process integration, Technical report," 2005.

[77] A. Macario, T.S. Vitez, B. Dunn, and T. McDonald, "Where are the costs in perioperative care? Analysis of hospital costs and charges for inpatient surgical care," *Anesthesiology,* vol. 83, pp. 1138-1144, 1995.

[78] D. A. Etzioni, J.H. Liu, M.A. Maggard, and C. Y. Ko., "The aging population and its impact on the surgery workforce," *Annals of Surgery,* vol. 238, pp. 170-177, 2003.

[79] A. C. F. E. Medicine, "Minimum standards for transport of critically ill patients," *Emergency Medicine,* vol. 15, pp. 202–204, 2003.

[80] C. S. Pattichis, E. Kyriacou, S.Voskarides, M. S. Pattichis, R. Istepanian, and C. N. Schizas, "Wireless telemedicine systems: an overview," *IEEE Antennas Propag. Mag,* vol. 44, pp. 143-153, 2002.

[81] H. Gholam Hosseini, M. Harrison, A. Al-Jumaily, and D. Benjamin, "Wireless Diagnostic tool for Anesthesiologists," in *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* Vancouver, British Columbia, Canada, August 20-24 2008.

[82] S. P. Nelwan, T. B. Van Dam, P. Klootwijk, and S. H. Meij, "Ubiquitous mobile access to real-time patient monitoring data," *Computing Cardiology,* vol. 29, pp. 557–560, 2002.

[83] B. Woodward, R. S. H. Istepanian, and C. I. Richards, "Design of a telemedicine system using a mobile telephone," *IEEE Transactions on Information Technology in Biomedicine,* vol. 5, pp. 13–15, 2001.

[84] K. Hung and Y. T. Zhang, "Implementation of a WAP-based telemedicine system for patient monitoring," *IEEE Transactions on Information Technology in Biomedicine,* vol. 7, pp. 101–107, 2003.

[85] P. Giovas, D. Papadoyannis, D. Thomakos, D. Soulis, C. Stamatopoulos, S. Mavrogeni, N. Katsilambros, G. Papazachos, and M. Rallidis., "Transmission of electrocardiograms from a moving ambulance," *Journal of Telemedicine Telecare,* vol. 4, pp. 5-7, 1998.

[86] S. Pavlopoulos, E. Kyriacou, A. Berler, S. Dembeyiotis, and D. Koutsouris, "A novel emergency telemedicine system based on wireless communication technology—AMBULANCE," *IEEE Transactions on Information Technology in Biomedicine,* vol. 2, pp. 261–267, 1998.

[87] B. Gohil, "Diagnostic Alarms in Anaesthesia," in *Engineering*: Auckland University of Technology (AUT), Auckland, NZ, pp. 132, 2007.

[88] S. K. Mitra, *Digital Signal Processing*, Second ed., 2001.

[89] G. D. Bergland, "A guided tour of the fast Fourier transform," *IEEE Spectrum* vol. 7, pp. 41-52, 1969.

[90] E. C. Ifeachor, *Digital Signal Processing*, Second ed. London: Prentice Hall, 2002.

[91] S. C. Sciacca and W. R. Block, "Advanced SCADA concepts," *IEEE Computer Applications in Power,* vol. 8, pp. 23-28, Jan. 1995.

[92] R. H. McClanahan, "SCADA and IP: is network convergence really here?," *Industry Applications Magazine, IEEE,* vol. 9, pp. 29 - 36, March-April 2003.

[93] J. D. McDonald, "Developing and defining basic SCADA system concepts," in *37th Annual Conference on Rural Electric Power Conference*, pp. B3/1 - B3/5, 25-27 April 1993.

[94] Q. Bin, G. Hoay, L. Yilu, and C. Eng, "Internet-based SCADA display system," *IEEE on Computer Applications in Power,* vol. 15, pp. 14-19, Jan. 2002.

[95] R. H. McClanahan, "The Benefits of Networked SCADA Systems Utilizing IPEnabled Networks," in *IEEE Rural Electric Power Conference*, pp. C5.1 - C5.7, 2002.

[96] "IEEE Recommended Practice for Data Communications between Remote Terminal Units and Intelligent Electronic Devices in a Substation," *IEEE Std 1379-2000 (Revision of IEEE Std 1379-1997),* 21 September 2000.

[97] K. Curtis, "DNP3 Protocol Primer," *DNP User Group,* 1 June 2000.

[98] G. A. Office, "Critical Infrastructure Protection Challenges in Securing Control Systems," October 1, 2003.

[99] J. Pollet, "SCADA Security Strategy," Plant Data Technologies August 8, 2002.

[100] L. A. Zadeh, "Fuzzy Sets," *Information and Control,* vol. 8, pp. 338-353, 1965.

[101] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Transactions on Systems, Man and Cybernetics SMC,* vol. 3, pp. 28-44, 1973.

[102] L. A. Zadeh, "On the analysis of large scale systems, Systems Approaches and Environment Problems," H. Gottinger, Ed. Gottingen: Vandenhoeck and Ruprecht, pp. 23-37, 1974.

[103] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *information and science,* vol. part-1, pp. 199-249, 1975.

[104] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems* vol. 1, pp. 3-28, 1978.

[105] D. T. Guy and Z. Slawomir, "The application of fuzzy logic and soft computing in information management " *Fuzzy Sets and Systems,* vol. 160, pp. 2117-2119, 2009.

[106] L. A. Zadeh, "Is there a need for fuzzy logic?," *Information Sciences,* vol. 178, pp. 2751-2779, 2008.

[107] L. A. Zadeh, "The birth and evolution of fuzzy logic " *International Journal of General Systems,* vol. 17, pp. 95-105, 1990.

[108] L. A. Zadeh, "Fuzzy logic and the calculi of fuzzy rules and fuzzy graphs," *Multiple-Valued Logic,* vol. 1, pp. 1-38, 1996.

[109] L. A. Zadeh, "Computing with words (CW) and its application to decision support and systems analysis," in *IEEE International Symposium on Intelligent Signal Processing,* pp. 1-2, 2003.

[110] L. A. Zadeh, "Soft computing and fuzzy logic," *Software, IEEE,* vol. 11, pp. 48-56, 1994.

[111] L. A. Zadeh, "Generalized theory of uncertainty--principal concepts and ideas," *Computational Statistics & Data Analysis,* vol. 51, pp. 15-46, 2006.

[112] J. S. R. Jang, "ANFIS: Adaptive-Network-based Fuzzy Inference Systems," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 23, pp. 665-685, May 1993.

[113] J. S. R. Jang and S. C. T., "Neuro-fuzzy modeling and control," in *Proceedings of the IEEE,* March 1995.

[114] E. H. Mamdani and A. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies,* vol. 7, pp. 1-13, 1975.

[115] E. H. Mamdani, "Advances in the linguistic synthesis of fuzzy controllers," *International Journal of Man-Machine Studies,* vol. 8, pp. 669-678, 1976.

[116] M. Sugeno, *Industrial applications of fuzzy control*: Elsevier Science Pub. Co., 1985.

[117] J. C. Bezdec, "Pattern Recognition with Fuzzy Objective Function Algorithms," New York: Plenum Press, 1981.

[118] S. Chiu, "Fuzzy Model Identification Based on Cluster Estimation," *Journal of Intelligent & Fuzzy Systems,* vol. 2, Spet. 1994.

[119] Mathworks, "MATLAB," R2008b ed USA, 2008.

[120] A. J. Asbury and Y. Tzabar, "Fuzzy logic: New ways of thinking for anaesthesia," *British Journal of Anaesthesia,* vol. 75, pp. 1-2, 1995.

[121] M. Akay, "Detection and Estimation Methods for Biomedical Signals," San Diego: Academic Press, 1996.

[122] I. Korhonen, J. Ojaniemi, K. Nieminen, M. van Gils, A. Heikela, and A. Kari, "Building the IMPROVE data library," *IEEE Engineering in Medicine and Biology,* vol. 16, pp. 21-24, 1997.

[123] L. R. M. Nieminen K., Morgan C.J., Takala J. and Kari A., "A clinical description of the IMPROVE data library," *IEEE Engineering in Medicine and Biology,* vol. 16, pp. 25-32, 1997.

[124] M. J. Harrison, *The enhancement of intra-operative diagnostics and decision-making using computational methods*. Auckland: University of Auckland, 2005.

[125] H. L. Kundel and M. Polansky, "Measurement of observer agreement," *Radiology,* pp. 303-308, 2003.

[126] M. Wolf, M. Keel, and S. von, "Improved monitoring of preterm infants by fuzzy logic," *Technol Health Care,* vol. 4, pp. 193-201, 1996.

[127] C. Oberli, J. Urzua, and C. Saez, "An expert system for monitor alarm integration," *Journal of Clinical Monitoring and Computing,* vol. 15, pp. 29-35, 1999.

Appendix A – Patient's Sample file (First Half Hour Data)

Rtime	Heart Rate (ecg.hr)	Average (5min)	HR-Avg (5min)	HR-SD	BP	BP\100	Average (5min)	BP-Avg (5min)	BP-SD	PV	PV-Avg (5min)	PV-AVG (5min)	PV-SD
3/2/2007 8:59	71		-0.83	0.71	10555	105.55		17.88	32.80	1108		49.82	32.53
3/2/2007 9:00	70		-1.83	0.71	5916	59.16		-28.51	27.29	1062		3.82	12.02
3/2/2007 9:00	71		-0.83	0.71	9775	97.75		10.08	2.19	1079		20.82	18.38
3/2/2007 9:01	70		-1.83	0.71	9465	94.65		6.98	8.60	1105		46.82	68.59
3/2/2007 9:02	71		-0.83	1.41	10681	106.81		19.14	34.87	1008		-50.18	34.65
3/2/2007 9:02	73		1.17	0.00	5749	57.49		-30.18	26.39	1057		-1.18	26.16
3/2/2007 9:03	73		1.17	0.00	9481	94.81		7.14	0.56	1020		-38.18	5.66
3/2/2007 9:03	73		1.17	0.71	9402	94.02		6.35	1.36	1028		-30.18	0.00
3/2/2007 9:04	74		2.17	0.71	9210	92.1		4.43	0.17	1028		-30.18	25.46
3/2/2007 9:04	73		1.17	1.41	9186	91.86		4.19	16.89	1064		5.82	12.02
3/2/2007 9:05	71		-0.83	0.71	6798	67.98		-19.69	15.52	1081		22.82	19.09
3/2/2007 9:05	72	71.83		0.71	8993	89.93	87.68		0.64	1054	1058.18		16.26
3/2/2007 9:06	73		1.9	0.71	9083	90.83		3.77	0.55	1077		7.89	0.71
3/2/2007 9:06	74		2.9	1.41	9005	90.05		2.99	1.46	1078		8.89	0.71
3/2/2007 9:07	72		0.9	0.71	8798	87.98		0.92	0.21	1077		7.89	2.83
3/2/2007 9:07	71		-0.1	1.41	8827	88.27		1.21	1.10	1081		11.89	18.38
3/2/2007 9:08	73		1.9	2.12	8671	86.71		-0.35	1.29	1055		-14.11	2.12
3/2/2007 9:08	70		-1.1	0.00	8854	88.54		1.48	2.02	1052		-17.11	7.07
3/2/2007 9:09	70		-1.1	0.00	8568	85.68		-1.38	0.40	1062		-7.11	8.49
3/2/2007 9:09	70		-1.1	0.71	8511	85.11		-1.95	0.97	1074		4.89	5.66
3/2/2007 9:10	69		-2.1	0.00	8374	83.74		-3.32	0.00	1066		-3.11	7.78
3/2/2007 9:10	69	71.1		0.71	8374	83.74	87.07		1.76	1055	1069.11		13.44
3/2/2007 9:11	68		1.46	0.71	8125	81.25		-4.96	50.39	1074		40.6	19.80
3/2/2007 9:11	69		2.46	0.71	15251	152.51		66.3	50.69	1046		12.6	14.85

3/2/2007 9:12	68		1.46	0.71	8082	80.82		-5.39	0.13	1067		33.6	18.38
3/2/2007 9:12	67		0.46	0.71	8064	80.64		-5.57	0.15	1041		7.6	2.12
3/2/2007 9:12	66		-0.54	0.71	8043	80.43		-5.78	0.45	1044		10.6	8.49
3/2/2007 9:13	67		0.46	0.71	7979	79.79		-6.42	0.38	1032		-1.4	7.07
3/2/2007 9:13	66		-0.54	0.71	8033	80.33		-5.88	1.48	1022		-11.4	2.12
3/2/2007 9:14	65		-1.54	0.00	7824	78.24		-7.97	0.18	1019		-14.4	18.38
3/2/2007 9:14	65		-1.54	0.71	7799	77.99		-8.22	0.49	993		-40.4	2.12
3/2/2007 9:15	66		-0.54	0.71	7868	78.68		-7.53	0.69	996		-37.4	72.12
3/2/2007 9:15	65	66.55		0.00	7771	77.71	86.22		0.86	1098	1033.40		9.90
3/2/2007 9:16	65		4.7	0.00	7892	78.92		-21.123	0.42	1084		167.13	11.31
3/2/2007 9:16	65		4.7	0.71	7951	79.51		-20.533	1.27	1068		151.13	26.87
3/2/2007 9:17	64		3.7	0.71	7771	77.71		-22.333	0.92	1030		113.13	12.02
3/2/2007 9:17	63		2.7	0.00	7901	79.01		-21.033	29.25	1013		96.13	239.71
3/2/2007 9:18	63		2.7	2.83	12038	120.38		20.337	3.08	674		-242.87	59.40
3/2/2007 9:18	59		-1.3	2.12	12474	124.74		24.697	6.09	758		-158.87	58.69
3/2/2007 9:19	56		-4.3	0.71	11613	116.13		16.087	4.77	841		-75.87	18.38
3/2/2007 9:19	55		-5.3	0.71	10938	109.38		9.337	1.79	867		-49.87	4.24
3/2/2007 9:20	56		-4.3	0.71	10685	106.85		6.807	0.67	861	916.875		9.90
3/2/2007 9:20	57	60.3		0.00	10780	107.8	100.04		2.31	875		19.4	6.36
3/2/2007 9:21	57		-0.5	0.00	10453	104.53		-1.08	0.26	884		28.4	7.07
3/2/2007 9:21	57		-0.5	0.71	10490	104.9		-0.71	1.46	874		18.4	4.95
3/2/2007 9:22	58		0.5	0.00	10697	106.97		1.36	0.67	867		11.4	7.07
3/2/2007 9:22	58		0.5	0.00	10602	106.02		0.41	0.45	877		21.4	12.02
3/2/2007 9:23	58		0.5	0.00	10539	105.39		-0.22	0.59	860		4.4	11.31
3/2/2007 9:23	58		0.5	0.00	10623	106.23		0.62	0.16	844		-11.6	21.21
3/2/2007 9:24	58		0.5	0.71	10601	106.01		0.4	0.11	814		-41.6	15.56
3/2/2007 9:24	57		-0.5	0.00	10585	105.85		0.24	0.23	836		-19.6	7.78
3/2/2007 9:25	57		-0.5	0.00	10552	105.52		-0.09	0.56	825		-30.6	9.90
3/2/2007 9:25	57	57.5		0.00	10473	104.73	105.62		1.55	811	855.6		2.12

Appendix B – Offline Analysis of Complete Data Using SMS

OFFLINE ANALYSIS of SCADA MONITORING SYSTEM (SMS)						AGREEMENT		DISAGREEM-ENT	
Patient No.	Time	Expert Diagnosis	MILD	MODER-ATE	SEVERE	TRUE POS SMS +ve Expert +ve	TRUE NEG SMS -ve Expert - ve	FALSE POS SMS +ve Expert -ve	FALSE NEG SMS -ve Expert +ve
Patient-5	0800								
	0815								
	0830								
	0845								
	0900								
	0915	N		P				1	
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	N					1		
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145								
	1200								
Patient-8	0800								
	0815								
	0830								
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		

Patient-11	0945	N					1		
	1000	N					1		
	1015	N	P					1	
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145								
	1200								
	0800								
	0815								
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	N		P				1	
	1030	P			P	1			
	1045	P			P	1			
	1100	P			P	1			
	1115	P			P	1			
	1130	P		P		1			
	1145	P		P		1			
	1200	P		P		1			
	1215	P	P			1			
	1230	N	P					1	
	1245	N					1		
	1300								

Patient-14	0800								
	0815								
	0830								
	0845								
	0900	P							1
	0915	N					1		
	0930	P		P		1			
	0945	P			P	1			
	1000	P			P	1			
	1015	P		P		1			
	1030	N	P					1	
	1045	N					1		
	1100								
Patient-18	0800								
	0815	N					1		
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	P		P		1			
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145	N					1		
	1200	N					1		

Patient-20	1215	N					1		
	1230	N		P				1	
	1245	P		P		1			
	1300	P		P		1			
	1315	N					1		
	1330	N					1		
	1345	N					1		
	1400	P	P			1			
	1415	P	P			1			
	1430	N					1		
	1445	N					1		
	1500								
	0800								
	0815								
	0830								
	0845								
	0900								
	0915								
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	P		P		1			
	1030	P		P		1			
	1045	P		P		1			
	1100	P			P	1			
	1115	P			P	1			
	1130	V			P	1			
	1145	V		P		1			
	1200	N		P				1	
	1215	N		P				1	
	1230	P	P	P		1			

	1245	N	P					1	
	1300	V			P	1			
	1315	N					1		
	1330	N					1		
	1345	N					1		
	1400	P	P			1			
	1415	P	P			1			
	1430	P	P			1			
	1445	N					1		
	1500	N					1		
	1515	N					1		
	1530	N					1		
Patient-21	0800								
	0815								
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N	P					1	
	1000	P		P		1			
	1015	N					1		
	1030	N					1		
	1045	N					1		
	1100	N		P				1	
	1115	N					1		
	1130	N					1		
	1145	N			P		1		
	1200	P			P	1			
	1215	P		P		1			
	1230	P	P	P		1			

	1245	P	P			1			
	1300		P					1	
	1315		P					1	
	1330								
	1345								
	1400								
Patient-24	0800								
	0815								
	0830								
	0845								
	0900								
	0915								
	0930	N					1		
	0945	N		P				1	
	1000	P		P		1			
	1015	N	P					1	
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115								
	1130								
Patient-27	0800								
	0815								
	0830								
	0845								
	0900								
	0915								
	0930								
	0945								
	1000								

	1015								
	1030								
	1045								
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145	N					1		
	1200	N					1		
	1215	N					1		
	1230	N					1		
	1245	N		P				1	
	1300	N					1		
	1315	N					1		
	1330	N		P				1	
	1345	P			P	1			
	1400	P			P	1			
	1415	N	P					1	
	1430	N					1		
	1445	N					1		
	1500	N					1		
	1515	N					1		
	1530	N	P					1	
	1545	N					1		
	1600	N					1		
Patient-29	0800								
	0815								
	0830								
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		

	0945	N					1		
	1000	N	P					1	
	1015	N					1		
	1030	N					1		
	1045	N		P			1		
	1100	P			P	1			
	1115	P		P		1			
	1130	N					1		
	1145	N					1		
	1200	N					1		
	1215	N					1		
Total						39	103	20	1

Appendix C – Offline Analysis of Complete Data using FLMS

	OFFLINE ANALYSIS of FUZZY LOGIC MONITORING SYSTEM (FLMS)					AGREEMENT		DISAREEMENT	
Patient No.	Time	Expert Diagnosis	MILD	MODE-RATE	SEVERE	TRUE POS FLMS +ve Expert +ve	TRUE NEG FLMS -ve Expert -ve	FALSE POS FLMS +ve Expert -ve	FALSE NEG FLMS -ve Expert +ve
Patient-5	0800								
	0815								
	0830								
	0845								
	0900								
	0915	N	P					1	
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	N					1		
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145								
	1200								
Patient-8	0800								
	0815								
	0830								
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		

Patient-11	0945	N					1		
	1000	N					1		
	1015	N					1		
	1030	N	P					1	
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145								
	1200								
	0800								
	0815								
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N	P					1	
	1000	N	P					1	
	1015	N					1		
	1030	P		P	P	1			
	1045	P			P	1			
	1100	P			P	1			
	1115	P			P	1			
	1130	P		P	P	1			
	1145	P	P	P		1			
	1200	P	P			1			
	1215	P							1
	1230	N					1		
	1245	N					1		
	1300								

Patient-14	0800								
	0815								
	0830								
	0845								
	0900	P							1
	0915	N					1		
	0930	P	P	P		1			
	0945	P		P		1			
	1000	P		P		1			
	1015	P		P		1			
	1030	N					1		
	1045	N					1		
	1100								
Patient-18	0800								
	0815	N					1		
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N					1		
	1000	N					1		
	1015	P		P		1			
	1030	N		P				1	
	1045	N		P				1	
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145	N					1		
	1200	N					1		

Patient-20	1215	N					1		
	1230	N					1		
	1245	P		P		1			
	1300	P			P	1			
	1315	N					1		
	1330	N					1		
	1345	N					1		
	1400	P		P		1			
	1415	P		P		1			
	1430	N					1		
	1445	N					1		
	1500								
	0800								
	0815								
	0830								
	0845								
	0900								
	0915						1		
	0930	N					1		
	0945	N					1		
	1000	N	P					1	
	1015	P		P		1			
	1030	P		P		1			
	1045	P		P		1			
	1100	P			P	1			
	1115	P				1			
	1130	V			P	1			
	1145	V		P		1			
	1200	N					1		
	1215	N					1		
	1230	P		P		1			

	1245	N		P				1	
	1300	V		P		1			
	1315	N					1		
	1330	N					1		
	1345	N					1		
	1400	P	P			1			
	1415	P	P			1			
	1430	P							1
	1445	N					1		
	1500	N					1		
	1515	N					1		
	1530	N					1		
	1545								
	1600								
Patient-21	0800								
	0815								
	0830	N					1		
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		
	0945	N	P					1	
	1000	P			P	1			
	1015	N					1		
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145	N					1		
	1200	P			P	1			

	1215	P		P		1			
	1230	P		P		1			
	1245	P	P			1			
	1300		P					1	
	1315			P					
	1330								
Patient-24	0800								
	0815								
	0830								
	0845								
	0900								
	0915								
	0930	N					1		
	0945	N		P				1	
	1000	P		P		1			
	1015	N					1		
	1030	N					1		
	1045	N					1		
	1100	N					1		
	1115								
Patient-27	0800								
	0815								
	0830								
	0845								
	0900								
	0915								
	0930								
	0945								
	1000								
	1015								

	1030								
	1045								
	1100	N					1		
	1115	N					1		
	1130	N					1		
	1145	N					1		
	1200	N					1		
	1215	N					1		
	1230	N					1		
	1245	N					1		
	1300	N					1		
	1315	N					1		
	1330	N			P			1	
	1345	P			P	1			
	1400	P		P		1			
	1415	N	P	P				1	
	1430	N					1		
	1445	N					1		
	1500	N					1		
	1515	N					1		
	1530	N					1		
	1545	N					1		
	1600	N					1		
	1615								
Patient-29	0800								
	0815								
	0830								
	0845	N					1		
	0900	N					1		
	0915	N					1		
	0930	N					1		

	0945	N					1		
	1000	N					1		
	1015	N					1		
	1030	N					1		
	1045	N		P				1	
	1100	P		P		1			
	1115	P	P			1			
	1130	N	P					1	
	1145	N					1		
	1200	N					1		
	1215	N					1		
	1230								
Total						37	108	15	3