National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

Implementing and Evaluating a Vital Sign Monitoring System in an ICU

Osama Abu-Shihab
B. Eng., McGill University, 1993

Department of Electrical Engineering
McGill University
Montréal
March, 1996

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements of the degree of
Master of Engineering

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

ISBN 0-612-12106-2

Canada

Abstract

The rapid growth of medical sciences and technologies created the need of increased use of computers to address the recognized problems associated with information overload, and to help health care professionals provide better quality decisions.

This thesis presents the development, implementation, and evaluation of a real-time expert monitoring system (EMS) developed for the patient data management system (PDMS) of a pediatric intensive care unit. The objective of the EMS is to generate real-time warning signals in the event of life threatening patient conditions.

The research in this thesis concentrated on the analysis of the performance of the expert system in the intensive care enviromnent by monitoring several patients over a period of days. The results obtained were generally in agreement with the actual medical interpretations given by the health care professionals at the MCH. However, some false positive and false negative results were observed and these are discussed in the thesis.

Résumé

La croissance rapide de la science médicale et de la technologie ont nécessité l'augmentation de l'usage des ordinateurs pour solutionner le problème de la surcharge d'information et pour permettre aux professionels d'améliorer la qualité de leurs décisions.

Cette thèse présente le dévelopement, la mise en oeuvre, et l'évaluation d'un Système Expert de Surveillance (SES) faisant partie intégrante d'un système de gestion de données dans une unité de soins intensifs pédiatrique. L'objectif du système de surveillance est de générer des alarmes en temps réel en cas de conditions critiques.

Cette thèse se concentre sur l'analyse des performances du système expert en milieu des soins intensifs. Les parametre vitaux de plusieurs patients furent surveillés pendant une période de plusieurs jours. Les résultats obtenus sont géneralment en accord avec les diagnostics du personnel medical du MCH. Cependant, quelque rèsultats faussement positifs et faussement negatifs furent obtenus, et sont analysés en detail dans ce mémoire.

Acknowledgments

I would like to extend my gratitude to Professor Alfred Malowany for his patience and understanding, and for his encouragement and guidance during my research.

I would like to thank Dr. Frank Carnavale and Dr. Ron Gottesman of the Montreal Children's hospital for assisting me in my work and for their time spent in the evaluation of the results.

Thanks also to my colleagues in rooms 118, 119 and 121 for their help and kindness. I am grateful to have been a member of such a friendly group.

Finally, my deepest appreciation must go to two special people in my life: my beloved mother for her constant support and endless love, and to my precious fiancée who was always there for me.

The financial support of CEFRIO is gratefully appreciated.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Intensive care medicine requires timely, accurate, and integrated patient records to provide the highest quality patient care. Computerized patient records offer the best methods to achieve these needs. However, for optimal use of computers in the intensive care unit (ICU), there must be a harmonious collaboration between medical informaticists, physicians, nurses, and administrators. The future use of computers in intensive care will be evolutionary rather than revolutionary. In the coming few years computers will become commonplace in the clinical care process.

In this thesis, an expert monitoring system will be presented and evaluated as part of a patient data management system in an ICU environment. This chapter addresses two main areas. One discusses patient data management systems, giving the definition, outlining the functionality, and describing the reliability of such systems. The second section introduces medical expert systems, highlighting their importance, their flaws, and the need for their evaluation. The chapter ends with a thesis overview in the last section.

1.1 Computers in the ICU

Over the last twenty years, the paper chart has been the primary method of information management in intensive care units. However, this sort of record keeping and management suffers from well known limitations: tediousness, illegibility, errors, and lack of privacy [Clayton and Hripcsack, 1995]. In one article, Hammond et al. presented a study which showed that errors occurred at least once in twenty-five percent of handwritten paper chart records for each twelve-hour nursing shift [Hammond et al., 1991 (1)]. In a different study, they further indicated that nearly one

third of all errors in an ICU involved mistakes in charting or relaying information between shifts [Hammond et al., 1991 (2)]

Furthermore, Edelstien argued that it takes time to document the nursing process using conventional paper chart methods [Edelstien, 1990]. In his research, Meyer stated that documentation related tasks consume from forty to sixty percent of the nurses' time [Meyer, 1992], time away from direct patient care.

In addition to these drawbacks, the sheer quantity of raw data collected on a single critically ill patient can be overwhelming. If manual paper chart is used to evaluate, sort, and integrate this data, eventual complications will arise [Milholland, 1988].

Solutions to these problems are seen in the application of computer technologies to clinical information management. In their study on the use of computers in the ICU, Goss et al. identified several benefits including: time saving, data sharing, easier monitoring, improved legibility, better organization, quicker access to information, reduced errors, and less paper waste compared to manual systems [Goss et al., 1995]. These benefits translate to increased nursing time with the patients and thus better direct care. Similar studies were done by Staggers who pointed out the user accountability and efficiency by using computers in the ICU [Staggers, 1988]. Tachakra et al discussed the importance of computers in reducing nursing staff workload [Tachakra et al., 1990], while Kriewell and Long related computers with database management for data manipulation [Kriewell and Long, 1991]. The use of computers for decision making was also investigated by [Gardner, 1990], [Safran et al., 1990], [Bowes and Wilson, 1994], and [Clayton and Hripcsack, 1995].

In the next section, a discussion of the features and functions of computers in the ICU is presented. The reliability of such systems and their interaction with the users is also mentioned.

1.2 Patient Data Management Systems

1.2.1 Definition and Features

Patient data management system (PDMS) is a term applied to any advanced, clinically oriented computer system developed specifically for the intensive care environment and designed to perform several operations on the high-volume, high-frequency patient data in critical care settings [Diaz and Haudenschild, 1983]. PDMSs usually have a comprehensive database that addresses multiple patient care systems as well as physiological and non-physiological parameters related to patients.

In general, PDMSs accelerate the collection, processing, and presentation of data, and reduce the workloads of nurses and doctors [Smith, 1992]. However, the principle objective of a PDMS varies according to its intended application [Nenov et al., 1994]. Andreoli divides the PDMS into three categories [Andreoli, 1985]:

1. Patient monitoring computer systems
2. Medical information systems
3. Computer assisted diagnosis systems

Moritz identified six functions for which PDMSs may be used: patient care, resource allocation, personnel management, planning and policy making, education for patients and employees, and investigations related to nursing research and clinical evaluation [Moritz, 1990]. Furthermore, Gardner described four functions of computers in most ICU settings: physiological monitoring, communication of data in multiple hospital locations, management of medical records, and computerized aid in patient care decision making [Gardner, 1990].

In the intensive care environment, PDMS can collect, store, and retrieve data from bedside monitors, fluid balance data, laboratory results, nursing notes, and care plans

[Milholland, 1988]. The data management functions of PDMS improve the organization and presentation of data in ways that promote efficient and effective analysis. These functions often include data sharing, data review, multiple access, data integration, and data security.

By providing data sharing, updating the value of a given data item in a PDMS will result in the update being available to all users and programs through a single entry. This saves time, reduces the risk of transcription error, and ensures consistency. Improved data review and access accomplished by the use of databases, multiple video display units, and printouts of the PDMS data allows health care professionals to find past data rapidly without searching through pages of handwritten material that may often be disorganized and illegible. PDMS also maintain the integrity and consistency of stored data by constantly checking preset boundaries on numerical information, and by furnishing the user with checklists in the case of non-numerical entries. Automatically acquired data are also evaluated for the presence of alarm notification and waveform problems. This allows PDMSs to play an important role in decision making in present day ICU [Clayton and Hripcsack, 1995].

Aside from improving the direct care of the patient, PDMSs have the potential to help hospitals lower costs by changing practice patterns and treating patients in appropriate, less expensive settings [Thompson, 1995].

1.2.2 Dependability and User Interaction

Different studies in the literature associate the acceptability of software to its reliability [Burkes, 1991] [Scarpa, 1992]. Reliability is the probability that a particular device will function as required in a specified environment for a particular period of time. The notion of reliability is important for devices that must provide continuous service [Johnson and Aylor, 1988]. However, in addition to reliability, other factors that are

involved in the success of medical software include availability, safety, and accessibility [Knight, 1990]. Availability is the probability that a particular device will be able to provide service at a particular time, whereas safety is the property that a device will not cause harm by operating incorrectly. The last factor, accessibility, plays an important part in the immediate acceptance or rejection of a software product because it conveys the ease of use of the system in the long run.

Accessibility is related to the end-user through the user interface. Current literature on the computerization of the ICU indicates that many human factor considerations are relevant to the design of the user interface. Hence, user interface technology has undergone significant improvement in the past ten years [Loner, 1990], and medical informaticists pay close attention to user interfaces when designing PDMS [Paganelli, 1989].

Several schemes of user interaction have been introduced in medical systems including: keyboard interaction [Solingan and Shabot, 1988], touch screen interaction [King and Smith, 1990], mouse interaction [Larson, 1992], and speech recognition [Shiffman et al., 1991][Petroni et al., 1991]. Each of these methods has its advantages and disadvantages, and the attitudes of end users toward use of computers vary according to these different modes of interaction. However, different studies illustrate that the overall use of computers in ICU is viewed by nurses and physicians as a step toward improving patient care [Johnson, 1990][Stonham, 1991][Scarpa et al., 1992]. In a survey done by Goss et al. on point-of-care computer systems, 70% of respondents reported a positive interest in installing a PDMS. To justify such a system 98% indicated productivity improvement as the major factor [Goss et al. 1995]. In another survey carried out by Burkes, 95% of nurses agreed that computers provided an easier patient information accessing and retrieving tool, and 100% of nurses admitted that computers provide better legibility [Burkes, 1991].

These results demonstrate that health care professionals are aware of significant benefits to computerization and are ready to make changes needed to incorporate the innovation of patient care. One such innovation that promises to elevate the standard of care in present intensive care units is the expert system. The concept of medical expert systems is introduced and analyzed in the next section.

1.3 Expert Systems

1.3.1 Medical Expert Systems

Expert system technology is entering its fourth decade of evolution and has proven successful when used as an adjunct to human decision making [Peterson and Fisher, 1993]. Consequently, many patient data management systems incorporate medical expert systems as a support to clinical personnel. Some examples of such PDMS include MYCIN [Shortliffe, 1976], MEDAS [Ben-Basset et al., 1980], CADIAC [Kolarz and Addlessing, 1986], KUSIN-MEDICINE [Saranumi et al., 1991], and MENINGE [Francois et al., 1992 (1)].

The intensive care units of hospitals are considered as suitable environments for utilizing expert systems. As the collection of data from intensive care patients is constantly increasing, large amounts of data become available to clinical computer systems. This data can be exploited to assess patients' conditions and to evaluate the effectiveness of therapy. The application of expert systems provides a promising approach to help the physician in these tasks [Rennels and Miller, 1988]. Expert systems have the potential to increase the quality of care by reducing human error, providing more accurate analysis of the large amounts of available data, and by guiding the clinical analysis into a correct direction [Perdu and Luis, 1991]

In one study, Francois et al. compare the diagnosis of MENINGE, a medical expert system for the diagnosis of child meningitis, to those of seven senior specialists and seven young physicians. The study showed that the system performs like the group of experts, with 94.8% of the presented cases correctly diagnosed by MENINGE [Francois et al., 1993 (2)]. A similar result was found by Kieth et al. Their expert system, which supports clinical decision making during labor, produced results that were indistinguishable from the diagnosis of doctors [Kieth et al., 1992]. In yet another study, Lucas and Jenssens conclude that HEPAR, a medical expert system in the field of hepatology, was capable of reaching a correct conclusion in 82% of the 181 selected patients [Lucas and Jenssens, 1991].

In spite of these and other successful results, knowledge engineers are faced with a shortcoming of medical expert systems: the fact that methodologies available for building expert systems are only suitable for well-structured systems and/or restricted-problem domains, whereas medicine can be characterized as a domain of great complexity where the task of discovering or identifying the knowledge needed for building expert systems is not always straightforward [Saranummi et al., 1991]. Moreover, any knowledge obtained may not be complete [Jones, 1991].

Consequently, in almost all of the cases found in the literature, the prototype expert systems built served only as demonstrations of the capabilities and limitations of this technology. A good example is the Thyroid Hormone Result Interpreter which has been in continuous experimental clinical use since 1987 and which has since then been successfully updated twice. However, attempts to transfer it to other hospitals have so far failed [Saarinen et al., 1991].

This leads to the conclusion that before expert system technology techniques can be applied on a larger scale in medicine, the medical field must become more structured and standardized to facilitate knowledge acquisition, knowledge representation, and the transfer of applications from one environment to another. Similarly, on the

methodology side, more efficient tools must be developed to identify and describe the roles of expert systems in clinical routine.

Another obstacle that challenges medical expert systems is one that is common to all aspects of the medical field, from patients to doctors to lab results, namely uncertainty. Uncertainty is the central, critical fact about medical reasoning [Szolovits, 1995]. However, numerous research studies in the field of medical informatics are aimed at tackling uncertainty in medical expert systems, and a myriad of published papers introduce various methodologies that attempt to reduce its effect in decision making. Some of the paradigms used to reach this goal include: certainty factor theory [Dan and Dudek, 1992], probabilistic inferencing [Todd et al., 1993] and fuzzy logic [Hajek and Hermancova, 1995] [Stotts and Kleiner, 1995].

Despite these apparent drawbacks of medical expert systems, it is still considered a thriving technology with great future potentials to improve health care [Wick, 1992]. More and more medical expert systems are being developed and introduced in the hope that one day they will provide perfect support in medical decision making, or even replace human expertise in certain areas [Nykennen et al., 1991]. To achieve this goal, however, formal evaluation is required to guarantee the validity and correctness of such systems. The next section explores this issue.

1.3.2 Evaluation of Expert Systems

The rapid increase in the use of expert systems over the last few years, coupled with the gradual automation of knowledge intensive work, lead to the need of systematic and reliable evaluation techniques. Medical expert systems require comprehensive evaluation of their diagnostic accuracy at every stage of development. Without established evaluation methods, the usefulness of medical expert systems is limited. Acceptance in the clinical arena is contingent on the verification of diagnostic accuracy

first and foremost. Unfortunately, not many medical expert systems that are described in the literature have been rigorously tested or exposed to formal evaluation [Lundsgaarde, 1987]. This may be due to the confusion of what to test and how to test it [Wyatt and Spiegelhalter, 1990]. Typically, there is no standard of what is right or wrong in medicine since expert health care professionals may disagree among themselves [Miller and Sittig, 1990]. As a result, it is difficult to define a correct answer to compare with that made by medical expert systems [Rossi-Mori et al., 1990].

Suen et al. divide the evaluation process into two components: verification and validation. During verification, the expert system is viewed as a "glass box" and a check is made that each component meets its specifications. During validation, the expert system is viewed as a "black box" and observations are made for its responses to test data [Suen et al, 1990]

1.3.2.1 Knowledge Base Verification

An important component of an expert system is its knowledge base which contains rules and facts about a problem domain. One of the critical issues in developing reliable expert systems is the verification of the knowledge base.

Building a knowledge base is an incremental process that involves transferring expertise from the human expert, through the knowledge engineer, into the computer [Gupta and Prasad, 1988]. Problems may arise in any stage of this process leading to an inaccurate or unreliable expert system. Thus, it is highly important to test knowledge bases for completeness, correctness and consistency before the expert system is installed and used in medical practice. The test applied during verification involve: detection of redundant rules, detection of conflicting rules, detection of circular rules, and detection of subsumption [Nguyen, 1985]. However, traditional testing methods, with the human expert to run many test cases on the system is impractical since the potential

number of cases increases exponentially as input parameters increase [Keng and Bahill, 1990]. As a result, many tools have been developed to help verify consistency and completeness of knowledge bases.

Earlier pioneer work includes the knowledge base verifiers, such as ONCOCIN rule checker [Suwa et al., 1982] and CHECK [Nguyen et al., 1987] which were built for specific knowledge based systems. They used domain-specific information in the verification process.

Verification tools developed later were based on a variety of different approaches and were capable of detecting more subtle cases. The KB-Reducer [Ginsberg, 1988] adopted a technique called knowledge base reduction to detect inconsistency and redundancy. In COVADIS [Rousset, 1988] a logical approach was used to check for inconsistency, which depended critically on the use of a forward chaining inference engine. EVA [Chang et al., 1990] proposed three types of checking: structural, logical, and semantic, and therefore offered the most comprehensive detection.

Liu and Dillon proposed a numerical Petri net to model the knowledge base of production rules. Reachability analysis was then conducted to reveal inconsistency and incompleteness in a knowledge base [Liu and Dillon, 1991]. In a more recent publication, Zhang and Nguyen introduced PREPARE: an automated tool for detecting potential errors in a knowledge base. PREPARE is based on modeling a knowledge base by using a Predicate/Transition net representation [Zhang and Nguyen, 1994].

1.3.2.2 Evaluation Schemes

Three major reasons exist for performing evaluation: ethical, legal, and intellectual [Wyatt and Spiegelhalter, 1990]. Evaluations aim at giving feedback to the developers and experts to assure that an expert system achieves the original requirements, the required quality, and the acceptable performance level [Nykennen et al., 1991].

Recently, many evaluation methodologies have been developed, and many publications can be found in the literature that deal with evaluation. However, these studies are spread across several disciplines, so their advice is not always applicable to the medical field. In addition, different sets of evaluating criteria exist depending on the purpose of evaluation [Liebowitz, 1986]. For example, Boehm et al. proposed an evaluation method concentrating on the software engineering aspect of expert systems. Their set of criteria included: portability, reliability, efficiency, testability, understandability, and modifiability [Boehm et al., 1978]. Gashing et al. however, emphasized usefulness in their method including: quality of the system's decision and advice, correctness of the reasoning techniques used, quality of the human computer interaction, system efficiency, and cost effectiveness [Gasching et al., 1983].

The issues of evaluating medical expert systems arose during the MYCIN development in the problem area of meningitis [Yu et al., 1979]. The evaluation of MYCIN was conducted based on the idea from Turing [Turing, 1963]. The study was a comparison of experts versus the system and conducted as a double blind trial where a set of outside experts evaluated the opinions of the experts and the MYCIN system. The same evaluation method was applied on INTERNIST-1 [Miller et al., 1982] and on ONCOCIN [Shortliffe et al., 1981]. A modified version of the Turing test, in which the system's results are compared with known standards instead of human performances, was used to determine the effectiveness of EMERGE, an expert system for chest pain analysis [Hudson et al., 1984]

However, the results of the above mentioned methods largely depended on the choice of test cases, and their accuracy relied on the number of test cases chosen. As an attempt to replace the unfeasible exhaustive testing, Miller and Sittig proposed and used three types of cases in a dynamic evaluation of HT-ADVISOR, an expert system developed to critique the pharmacological management of essential hyper-tension. First, a large set of real cases was obtained from the clinical environment. Second, a comprehensive set of artificial cases was constructed specifically to challenge the system's response both to clinically important and to unusual sets of conditions. Finally, a robust set of cases was gathered from the domain expert's own practice [Miller and Sittig, 1990].

In a more recent study, Georgeakis et al. applied sensitivity, specificity, and system response measures to provide a statistical evaluation methodology to assess the performance of medical expert systems including MEDAS, the Medical Emergency Decision Assistance System [Georgeakis et al., 1991]. Their study showed that the same measures used by the social sciences to examine the performance of human experts in the decision making process can be used in evaluating medical artificial intelligence systems.

The evaluation of medical expert systems has always been an important part of implementation. It is a necessary task to perform before taking any system into routine use. Yet, neither single nor global methodologies exist which covers all the particular problems related to the decision support systems in medicine [Clayton, 1995]. Nevertheless, different methods and techniques may be combined to reach the desired evaluation strategy. It is important to bear in mind, however, that decision support systems are mainly intended as supporting tools for end-users, therefore evaluation should eventually measure the quality characteristics of the user-system integrated behavior and the user's performance with the system. Thus expert systems should always be evaluated as part of the environment, not as stand alone devices.

1.4 Thesis Overview

In this thesis, an expert monitoring system is presented as part of a patient data management system in an intensive care environment. Chapter two begins by describing the overall PDMS system along with its hardware and software architectures. A brief overview of the different modules involved in its creation is also outlined.

The first part of chapter three presents the conceptual description and design of the expert monitoring system. The second part focuses on the implementation procedures of the different parts of the EMS and their integration to produce the desired results.

The evaluation of the EMS, including laboratory and field tests, together with the results obtained are discussed in chapter four. Possible future work and extensions are highlighted before concluding with chapter five.

Chapter 2 The Patient Data Management System

This chapter introduces the patient data management system (PDMS), its objective, its architecture, and its different modules. The system is being developed at McGill University in coordination with the Montreal Children's Hospital for the eventual use of the system at the hospital's intensive care unit.

2.1 PDMS Design Objective

The PDMS is a real time medical information system designed to analyze clinical data, develop adaptive interpretations and elaborate real-time therapeutic guidelines for the management of ICU patients. The system also aims to meet the needs of department managers, hospital managers and health care authorities.

The PDMS offers fast processing capabilities as well as large storage capacity on data it automatically collects from patients' bedside monitors. It also furnishes record keeping quality for both clinical and managerial use. An expert monitoring system and a nursing scheduler reduce the workload on the attending health care professionals by providing emergency warning signals and computerized to-do lists, respectively. User friendly interfaces and graphical displays on color video monitors are used to present the information of the PDMS in a simple and direct way.

2.2 The Hardware Architecture

The main component of the PDMS is the Hewlett-Packard (HP) CareNet System. This system supports a local area network (LAN) in a star configuration. At the center of this network is a HP78581A Network Communications Controller connected to

fourteen HP78534A Physiological Bedside Monitors. The monitors are capable of measuring patients' vital signs, smoothing measured parameters and generating alarms signals. A host computer, an Intel 80486-based microcomputer, communicates with the network through an RS-232 serial link and a HP78588A Careport interface, which translates the patient's data form HP format into standard RS-232 format. Figure 2.1 shows the PDMS hardware architecture.

The host computer has 16 Megabytes of RAM, and a 300 Megabytes hard disk. Information is displayed on a high resolution, color display adapter that provides a resolution of 1024x768 pixels. A future extension of the architecture will connect the host to other computers by linking them in a Token-Ring LAN.

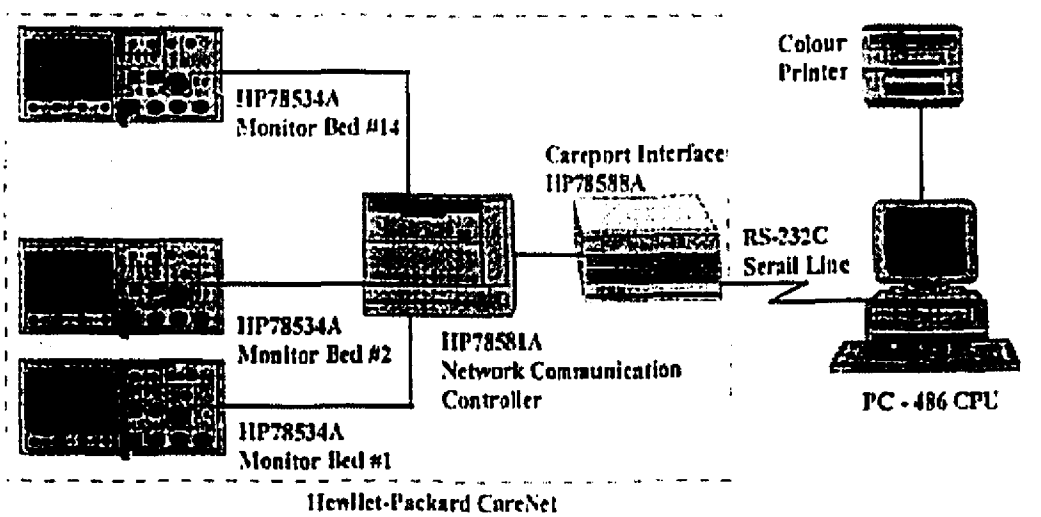

Figure 2-1 PDMS Hardware Architecture

2.3 The Software Architecture

The software architecture of the PDMS is modular in design. This is done to ensure that the system is maintainable, upgradable, and locally configurable. Moreover,

modularity permits a more efficient software, with separate programs implementing only necessary functions rather than one program for all tasks.

The OS/2 multitasking operating system is chosen as a platform to implement the different modules of the PDMS. With OS/2, multiple tasks and threads are readily available. Furthermore, shared memory segments, pipes, and semaphores provide interprocess communication between the modules. In addition, OS/2 provides an easy windowing interface to the PDMS, namely the Presentation Manager.

2.4 PDMS Modules

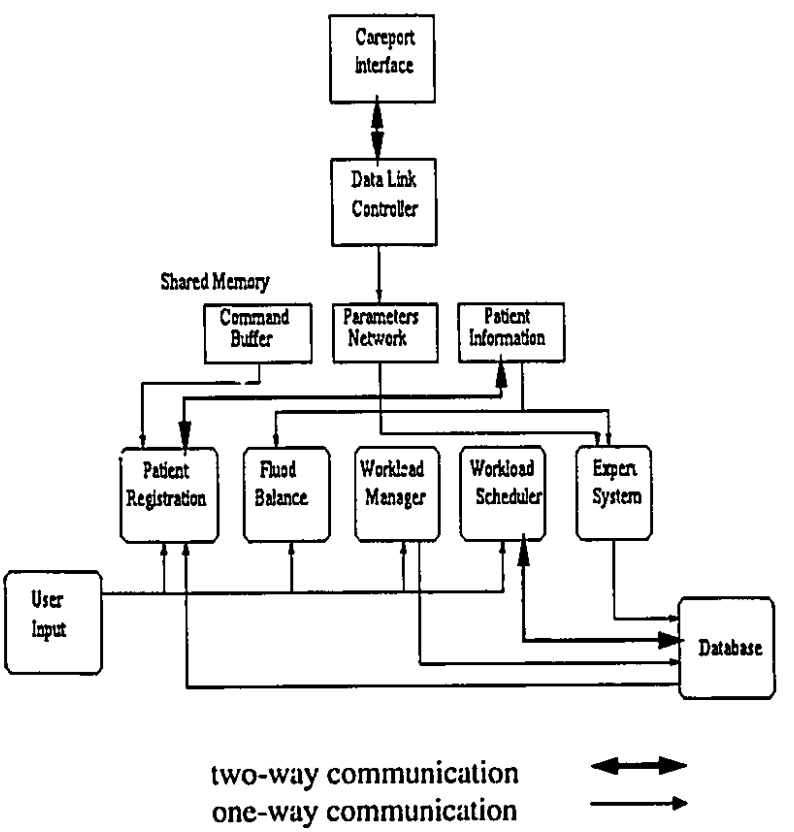

Figure 2-2 PDMS Software Configuration

Figure 2.2 shows the software configuration of the different PDMS modules.

2.4.1 The Data Link Controller

The primary function of the Data Link Controller (DLC) module is to acquire, in real-time, the physiological data generated by the bedside monitors and transmitted by the Careport via the RS-232 serial link. The data acquisition occurs every two seconds. However, this read data is averaged every minute and the results are placed in circular queues for each of the fourteen beds [Fumai et al, 1991].

The DLC also works in the opposite direction by transmitting signals from the modules to the CareNet system.

2.4.2 The Patient Registration Module

The Patient Registration Module manages the admission, suspension and discharge of patients in the intensive care unit. It also handles administrative patient information, such as name, sex, date of birth, bed assignment and so on, by allowing users to enter, modify and review patient information through a menu driven user interface. Figure 2.3 shows the editing dialog box of the registration module.

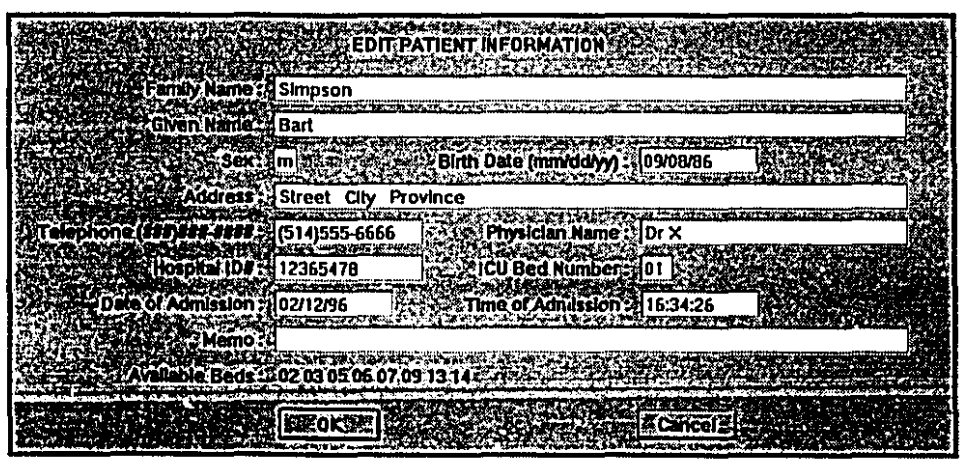

Figure 2-3 Dialog Box for Editing Patient Information

The entered patient information is automatically saved in the database module as well as in shared memory. Other PDMS modules, such as the Nursing Workload Scheduler and the Expert Monitoring System, can access the patient information through the shared memory mechanism. By storing the information in the database, a rapid and simple procedure can be implemented to reload the patient information in the case of application failure or system shutdown [Zia, 1996].

This module also plays a role in the acquisition of patients' vital signs. It signals to the DLC to start the process of acquisition once a patient is registered and a bed is occupied.

2.4.3 The Fluid Balance Module

The Fluid Balance module is responsible for recording the fluid intake (ingesta) and fluid output (excreta) of each patient in a spreadsheet format. Figure 2.4 shows the main ingesta window of the module. Periodic measurements from infusion pumps and urine bags are entered by the nursing staff to determine the overall balance of fluids.

A speech interface that allows direct entering of data by the use of voice is also available [Petroni et al., 1991]. This provides a hands-free capability and enhances the mobility of the user while entering data.

Any data entered into the spreadsheet is automatically saved into the PDMS database module.

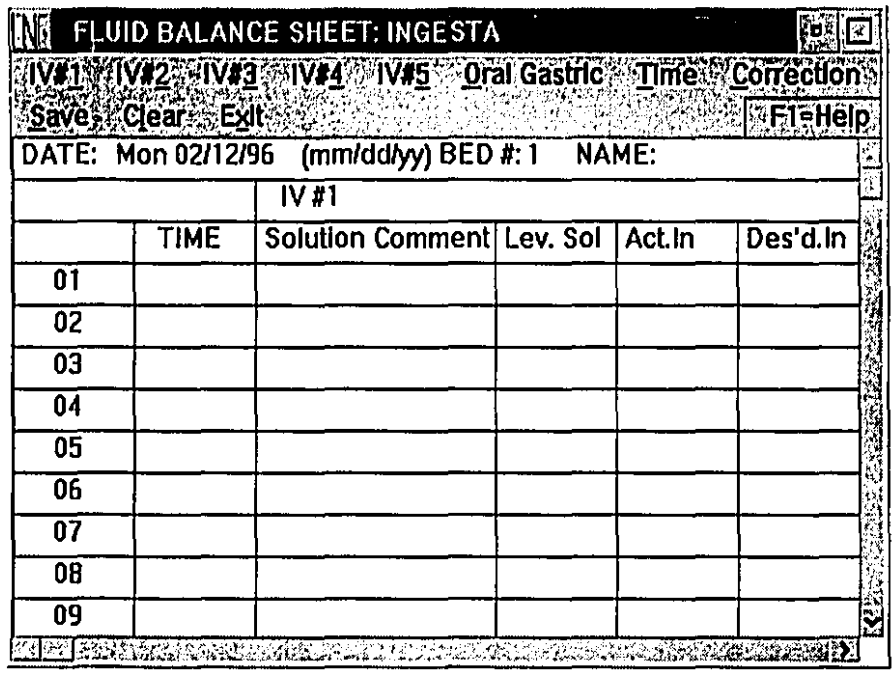

Figure 2-4 Main Window of the Fluid Balance Ingesta Module

2.4.4 The Nursing Workload Manager Module

The Nursing Workload Manager module is used to produce nursing care plans for each patient in the ICU. The module allows new care plans to be created manually or, as a second option, permits customizing a standard care plan according to a patient's situation. A library of care plans is pre-integrated into the module. It contains general-task standard care plans, as well as some diagnosis-specific ones. However, users can add new care plans to this library whenever required [Roger, 1992]. Figure 2.5 shows the main window of the nursing workload manager module.

The module also supports the PRN (Progressive Research in Nursing) workload measurement scoring system. The scoring provides feedback to the administration of the ICU used in determining staffing allocations, monitoring productivity and cost of nursing services.

Any data entered into the nursing care plan is automatically saved into the PDMS database module.

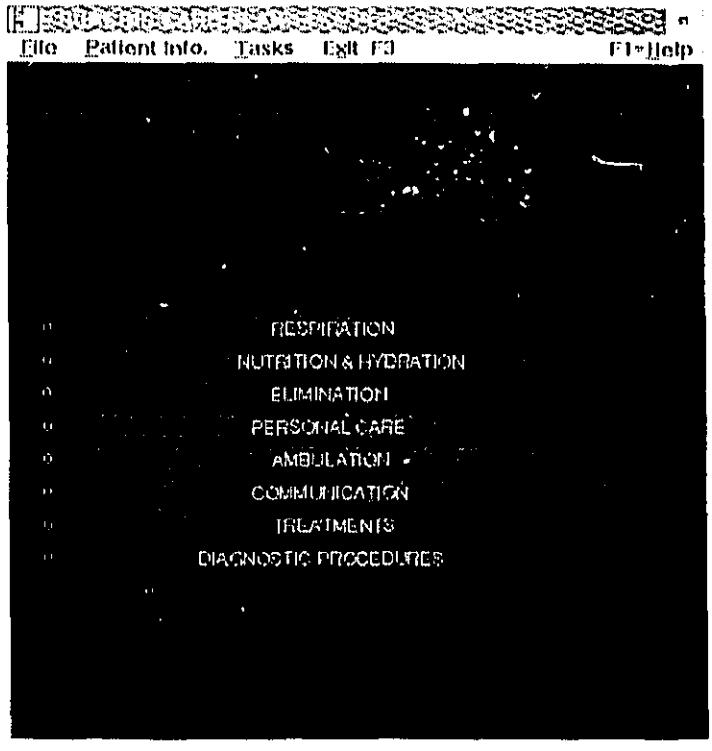

Figure 2-5 The Main Window of the Nursing Workload Manager

2.4.5 The Nursing Workload Scheduler Module

The Nursing Workload Scheduler module works in conjunction with the workload manager module. It consists of an expert system which can generate schedules for the different tasks to be carried out by the attending nurses of the ICU. The tasks submitted for scheduling come from the eight categories in the nurse workload manager module: respiration, nutrition and hydration, elimination, personal care, ambulation, communication, treatment, and diagnostic procedures.

By accessing the database module, the information required for the scheduling process is retrieved from the nursing care plans already created for each patient. However, the

user interface of the scheduler allows the user to add, delete or modify any of the tasks in the care plan [Sun, 1993]. Once sufficient information is provided, and the expert system properly initialized, the Nursing Workload Scheduler automatically resolves time conflicts and generates complete schedules. Figure 2.6 shows the main window of the module.

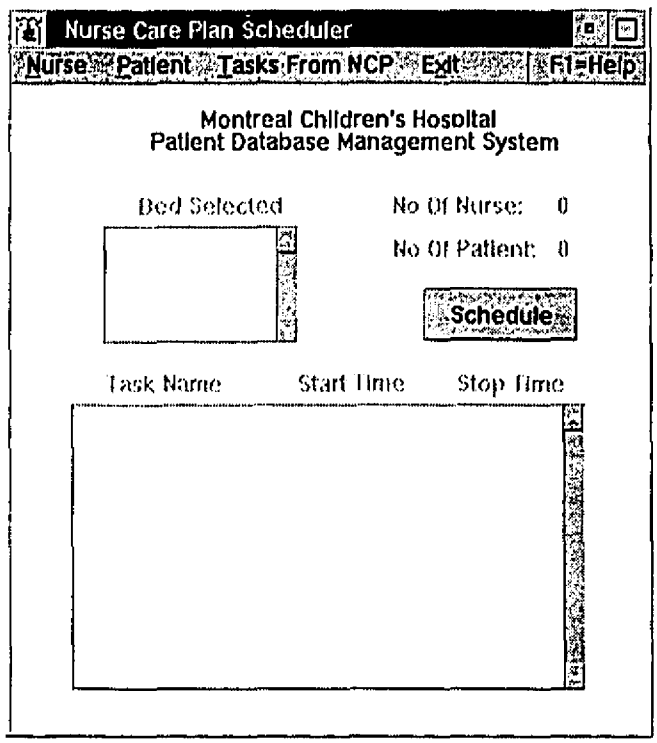

Figure 2-6 Main Window of the Nursing Workload Scheduler Module

2.4.6 The Database Module

The Database Module creates various database tables for the storage of data sets from the Patient Registration Module, Fluid Balance Module, Workload Manager Module and the Expert System Module. It also acts as the data input source for the Nursing Workload Scheduler Module, and in some cases for the Patient Registration Module as discussed in section 2.4.2.

The Module is implemented using the relational database management services available from OS/2 Extended Edition Database Manager [Saab, 1995]. Interaction with the Database Manager is performed via C Language and embedded SQL (Structured Query Language) statements

2.4.7 The Expert Monitoring System Module

The Expert Monitoring System (EMS) Module provides automated physiological trend analysis and interpretations in real-time. It also produces warning signals in critical circumstances for every patient in the ICU.

The EMS consists of linearization filters and an expert system. It receives the physiological data of each patient from the DLC module. After eliminating irrelevant information from this data, the EMS extracts linear trends for analysis. The knowledge-based expert system will then assess the present and near-future condition of the patient based on the examination of the linear trends. A user interface relays the information to the end user and generates alarms in cases of emergency. The vital signs and the output conditions of the expert system are saved into the PDMS database for future reference.

The implementation and evaluation of the EMS are presented in the following chapters.

Chapter 3 The Expert Monitoring System

3.1 The Expert Monitoring System

3.1.1 Conceptual Description

The goal behind the expert monitoring system (EMS) is to detect various patient conditions in real time, and to generate alarms in cases of critical situations. Health care professionals in the intensive care unit are faced with the difficult task of maintaining many records that cover specific areas of patient care and treatment. Assessing these records to enable the most effective clinical decision without any misinterpretations, specifically in stressful situations, is not easy [Bowes et al., 1994] [Brown et al., 1994]. The EMS acts as an aid to the nurses in the ICU by providing trend analysis and possible interpretations.

The EMS functionality is based on the cardiovascular system (CVS) which includes the heart, arteries, and veins. The values of the heart rate (HR), blood pressure (BP), and central venous pressure (CVP) can give a good evaluation of a patient's condition [Green, 1982]. The EMS focuses on the monitoring of these parameters. In addition to their importance, they are readily available by the automated data collection of the PDMS.

3.1.2 System Specifications

The structure of the EMS is composed of three different components organized in a pipeline as shown in figure 3.1: 1. linearization filters for noise removal; 2. an expert system for diagnosis and prediction; 3. a user interface.

The EMS is part of the PDMS and is connected to its front-end inputs via the DLC. It is also linked to the PDMS through the database module in which it stores its output interpretations.

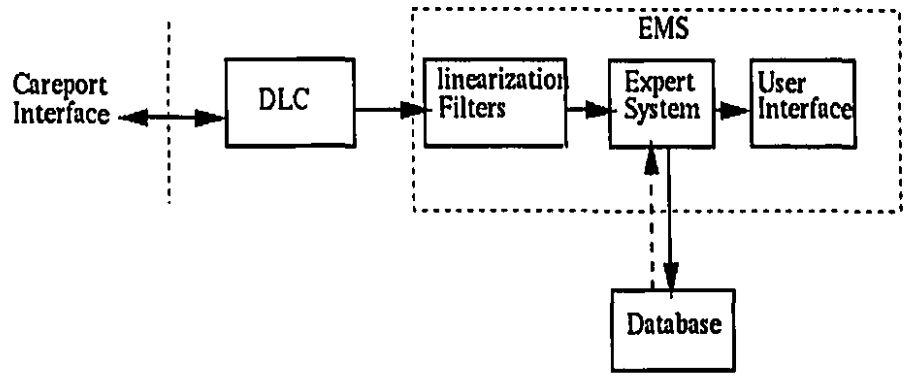

Figure 3-1 The structure of the EMS and its connection with the PDMS

The data link controller sends the values of the HR, BP, and CVP of each patient in the ICU every minute. The linearization filters receive this data and process it to extract linear trends and duly send them to the expert system. The knowledge based expert system uses production rules to conclude the current and near-future predicted conditions of the patients according to the linear trends it receives. At this point the user interface, which displays all of the ICU patients' state simultaneously, is updated. If a patient is deemed as being in critical situation by the expert system, color codes and alarms are generated. At this stage, information in the form of vital signs, conditions and time stamps of every patient are passed back to the PDMS and stored in the database tables. A possible extension on the expert system is shown as a dashed arrow in figure 3.1. This would allow a two way communication with the database, hence enabling a more refined system to incorporate drugs and laboratory results in the diagnosis [Autio et al., 1991].

3.1.3 Design

When describing the state of the patient, three types of uncertainties must be resolved by the EMS as shown in figure 3.2: 1. Measurement uncertainty of patient's parameters; 2. the uncertainty between the possible diagnosis based on the combination of the patient's parameters; and 3. the evolution uncertainty of the patient's state which changes with time.

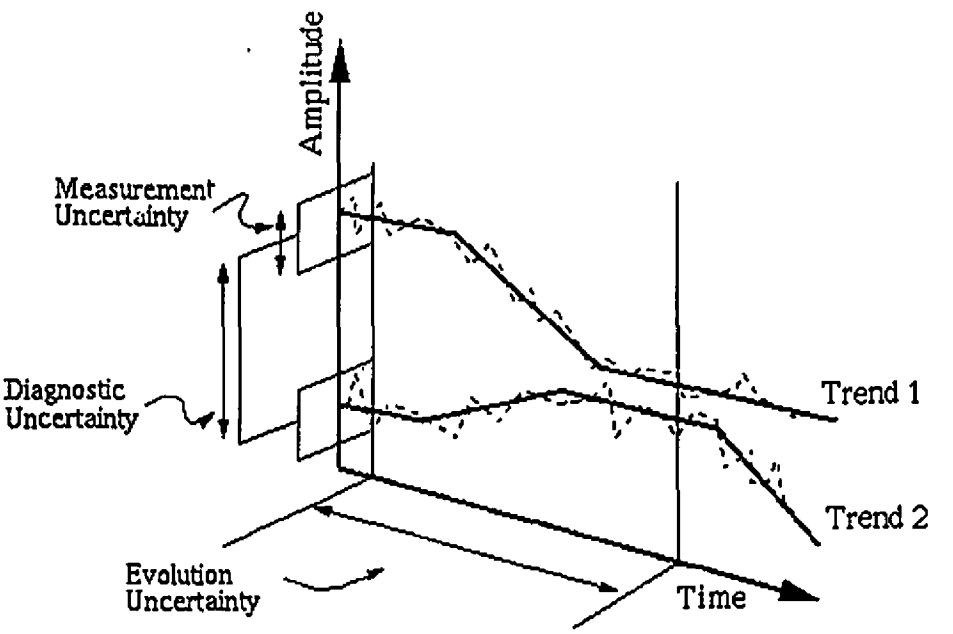

Figure 3-2 Sources of uncertainties in the diagnosis of the EMS

The measurement uncertainty is decreased by using filters and a linearization algorithm; the diagnostic uncertainty is resolved by a set of production rules; while the evolution uncertainty is minimized through the concept of certainty indices to be discussed shortly.

3.1.3.1 Linearization Algorithm

The different units of the linearization module are shown in figure 3.3.

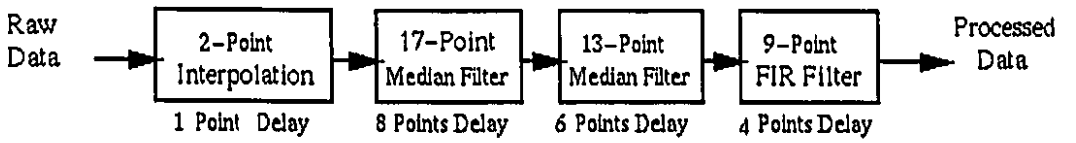

Figure 3-3 Different components of the linearization module

The design of the EMS is such that it would notify the attending nurses in alarming situations. However, the false alarm rate as well as the correct alarm rate are both dependent on the alarm limits. Minimizing the frequency of false alarms may be achieved by choosing wide limits sufficient only for the detection of significant changes in the patient's physiological status. However, this also decreases the frequency of useful correct alarms. On the other hand, choosing tight limits maximizes the number of correct alarms, but at the same time increases the number of false alarms due to brief, clinically insignificant fluctuations in parameter values [Koski et al., 1990]. The linearization module, with its four components, removes these fluctuations from slower developments.

Median filters have been used in a variety of fields in the context of noise removal [Esfahani et al., 1993] [Stothert et al., 1994]. In this design, two median filters are applied to preprocess the parameter values used in the expert system module. A similar design has been used by Makvitra et al. [Makvitra et al., 1991].

In the current implementation of the PDMS, the data link controller sends out the physiological parameter values of each patient every minute. When this data is received by the linearization module of the EMS, a 2-point interpolation algorithm interpolates the initial minute samples to half minute data. Two median filters of order 17 and 13 are then used to remove the unwanted impulses on the trend. However, a

major drawback of the median filter is its delay time [Gallagher ,1988]. The first 17-point median filter results in a delay of 8 points (4 minutes). To remove the artifacts created by the first pass of this long median filter, the 13-point filter is subsequently applied. A 9-point notch filter without any memory elements (FIR) is then applied as a context smoother. The coefficients of the transfer function are the result of convolving the factors $G_i[1-2\cos w_i z^{-1} z^{-2}]$ four times, twice with w_i equal to 0.75π and 0.5π respectively. The G_i is adjusted so that the zero gain is 1. The development of the linearization filters coefficients forms part of a separate study [Collet et al., 1992].

This design achieves a compromise between smoothing of the trends, for a good performance of the next expert system stage, and preserving the sharp trend changes as much as possible.

3.1.3.2 Conditions and Production Rules

After the physiological parameters have passed through the linearization module, they undergo further processing by the expert system module, the core of the EMS. In this module, a set of production rules are used to generate the best estimation of the patient's condition. The functional domain of the EMS consists of eleven patient conditions which are most commonly found in the ICU. They are:

- *Hypovolemia*: blood volume is too low through dehydration, hemorrhage.
- *Hypervolemia*: blood volume too high due to the uncareful monitoring of drugs and IVs.
- *Bradyarrhythmia*: the heart beats at an abnormally low rate.
- *Tachyarrhythmia*: the heart beats at an abnormally high rate.
- *Tamponnade*: a hemorrhage takes place around the heart reducing the space needed for it to fill up. The heart stroke volume is thus reduced.
- *Pump Failure*: failure of the heart muscle to pump blood correctly.

- *CNS-ICP*: increased Intra-Cranial Pressure (ICP) in the brain due to some kind of problem in the nervous regulation of the CVS.
- *CNS-Drugs*: abnormal behavior of the nervous regulation caused by some inhibitory effect of drugs.
- *Primary Hypertension*: high blood pressure.
- *Systemic Shock*: chemicals affecting the vasodilator receptors of the CVS, caused by a bacterial infection or an allergic overreaction.
- *Agitation*: the state of a patient under stress from pain, crying or restlessness.

As was mentioned earlier, the values of the HR, CVP, and BP give a good evaluation of the state of the patient. Thus, these particular physiological data are used to generate values that determine which of the eleven conditions, described above, a patient is suffering from. The tabular description of these rules, shown in table 3.1, will be explained in the following sections.

Since the EMS's goal is to monitor in real time, emphasis in the design of the expert system module is on responsiveness rather than detail of diagnosis. Hence very few rules have conclusions as conditions for other rules. Karp et al. described this to be a shallow rule set [Karp et al., 1988]. Nevertheless, this would enable the expert system to generate rapid responses. This is ensured by processing the physiological data in a pipeline structure and allowing no backtrack or revisions of previous solutions, rather the expert system provides estimates of immediate situations depending on the next upcoming values.

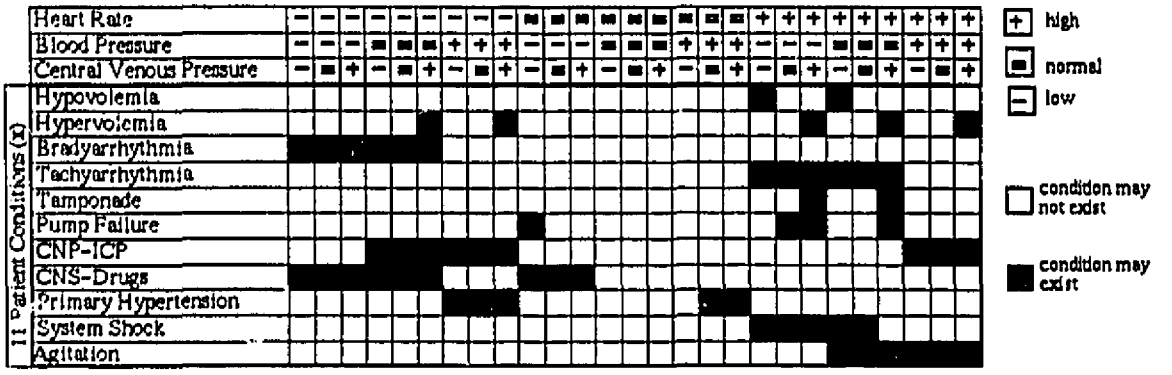

Table 3-1 Basic Diagnostic Rules

3.1.3.3 Rule Severity Level

Table 3.1 expressed the monitored parameters as high, normal or low for establishing the set of rules represented. A parameter becomes increasingly abnormal as its value deviates from the normal value. This is shown in figure 3.4.

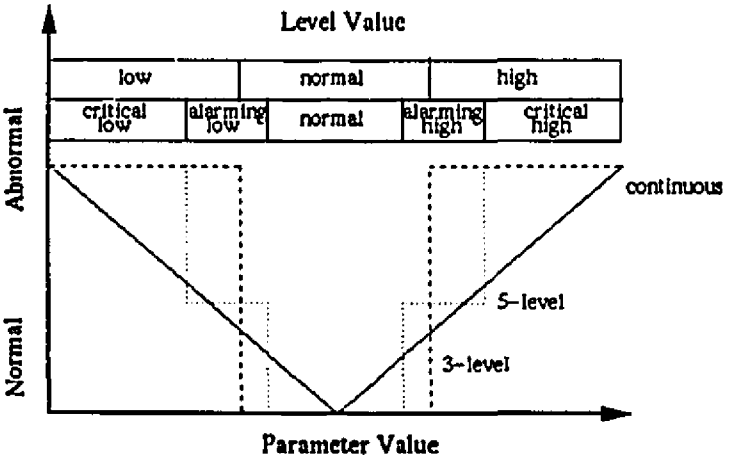

Figure 3-4 Translation between Parameter and Level Description

However, instead of using a simple 3-level description (low, normal, high), a 5-level solution based on table 3.1 is applied to better approximate the continuous case: the high and low levels are further subdivided into critical and alarming levels. Thus, if 0

were to represent the normal condition, −1 and −2 would represent alarming and critical low, while 1 and 2 would represent alarming and critical high, respectively. Although the exact values are not important, the relative values between the numerical representations carry some meaning.

The severity of the rule condition level is the maximum of the three contributing parameter levels. This means that a rule is considered 'critical' as soon as any of its parameters has this level. In other words, the rule condition level d of a patient condition x becomes:

$$d_x = \max(|HR|, |BP|, |CVP|) \quad \text{where } 0 \le d \le 2 \tag{3-1}$$

Since the EMS handles raw or prefiltered data stream as well as processed or filtered data, d_x^r and d_x^p from the respective data streams are then calculated according to equation 3.2:

$$d_x^r = \max(|HR^r|, |BP^r|, |CVP^r|) \;\&\; d_x^p = \max(|HR^p|, |BP^p|, |CVP^p|) \tag{3-2}$$

3.1.3.4 Condition Certainty Indices

The EMS estimates the possibility that a condition x is occurring by detecting its corresponding parameter patterns, shown in table 3.1, and calculating a condition certainty index (CI) that reflects the accumulated number of times condition x has occurred. The greater this number over a given period of time, the more probable that condition x is diagnosed. When a condition is not diagnosed, its corresponding certainty index is decreased. Thus, darkened entries in table 3.1 denote positive contributions towards the calculation of the certainty indices, while negative processing applies to all blank entries.

To compensate for the linearization delay produced by the filters, the certainty index is calculated using both the raw and processed data:

$$CI_x = \left\{ \begin{array}{ll} CI_{max}, & \text{if } CI_x > CI_{max} \\ CI_{min}, & \text{if } CI_x < CI_{min} \\ w^r \sum (w_+ d^r_{x+} - w_- d^r_{x-}) + w^p \sum (w_+ d^p_{x+} - w_- d^p_{x-}), & \text{otherwise} \end{array} \right\} \quad (3\text{-}3)$$

where

$$1 - w^p = w^r \quad (3\text{-}4)$$

$$1 - w_+ = w_- \quad (3\text{-}5)$$

p : processed data
r : raw data
x+ : positive diagnosis of condition *x*
x- : negative diagnosis of condition *x*
d : rule condition level (0, 1, or 2)

w^p : weight of contribution of processed data on CI
w^r : weight of contribution of raw data on CI
w_+ : weight of positive diagnosis
w_- : weight of negative diagnosis

The formula accumulates the summation over a series of actual data points obtained in real time. For darkened entries $d_{x-} = 0$ and $d_{x+} = d$ (of equation 3.2), and for blank entries $d_{x-} = d$ (of equation 3.2) and $d_{x+} = 0$. This applies to both the raw and processed versions. Weights are used to control the significance of raw (w^r) and processed (w^p) data, and the positive (w_+) and negative (w_-) diagnosis in calculating CI.

3.1.3.5 Prediction Certainty Indices

A similar diagnosis to the one carried out in the previous section can be performed on extrapolated patterns of parameter levels. This would present the EMS with an added advantage of predicting upcoming patient conditions. However, the prediction index is calculated with a different weighting scheme. Since prediction beyond a certain time

become unreliable, the weighting factor incorporates the maximum prediction time (t_{max}) and the time (t) of the next level crossing. This is done based on the current parameter value and slope. Thus,

$$w^*(t) = \begin{cases} 1, & if\ t \le 0 \\ 1 - \dfrac{t}{t_{max}}, & 0 \le t \le t_{max} \\ 0, & otherwise \end{cases} \qquad (3\text{-}6)$$

and the predicted rule severity level becomes:

$$d_x^* = \max(w^*(t_{HR})|HR_{next}|, w^*(t_{BP})|BP_{next}|, w^*(t_{CVP})|CVP_{next}|) \qquad (3\text{-}7)$$

where <parameter>$_{next}$ is the predicted (extrapolated) parameter level, and $t_{<parameter>}$ is the predicted level crossing time for that parameter. Prediction certainty indices (CI^*) are computed using equation 3.2 to 3.5 with d_x^* replacing d_x. Further discussion on the predictive weights expressed in the formula above can be found in [Lam, 1993].

3.1.4 User Interface

The user interface of the EMS displays all the fourteen beds of the ICU simultaneously. A color coding scheme is used on each bed. Red, yellow, and green colors are used to indicate critical, alarming and normal conditions respectively; while a black color denote vacant beds. If the certainty indices calculated by the expert system for a patient exceed the preset threshold values associated with any patient, the appropriate color is generated on screen for that bed. Clicking on any particular bed activates a window displaying the patients parameters in addition to the current and predicted states of each of the eleven possible conditions.

An important aspect in the design of the user interface is that it requires no data entry from the clinical personnel, consequently minimizing the time required to operate it.

3.1.5 Database Considerations

The EMS handles vital signs from the data link controller in real-time and produces warnings in cases of emergency. However, storing the vital signs and the patient conditions interpreted by the monitoring system is also needed for the following reasons:

1. Efficiently analyzing the EMS performance without extensive recomputing of the patient conditions.
2. Archiving of significant incidents occurring during the patient's stay in the ICU.
3. Evaluating the overall ICU management.

The current database module uses a relational database and forms part of the PDMS. Although the present design provides for dynamic logging of the patient's vital signs [Saab et al., 1995], the data stored is not sufficient for evaluation purposes. In order to evaluate the EMS, the required information is stored in the EMS Conditions Table shown below. In this table, the following twenty-seven variables are defined:

- Patient_ID: indicates the hospital ID number given to the patient. *Domain*: serial number. This object identifier should contain a unique not null value.
- HR_Value, BP_Value and CVP_Value: indicate the vital sign parameter value recorded at an instant in time. *Domain*: bit data type.
- C0_Value ... C10_Value and PC0_Value ... PC10_Value: indicate the condition and the predicted condition status recorded at an instant in time. *Domain*: bit data type.

- Time_Stamp: indicates the time when the parameter and the condition were measured. *Domain*: Time Stamp.

Attribute Name	Null	Unique	Domain	Bytes
Patient_ID	N	Y	id	20
HR_Value	N	N	bit data	dflt
BP_Value	N	N	bit data	dflt
CVP_Value	N	N	bit data	dflt
C0_Value	N	N	bit data	dflt
.	.	.	.	.
.	.	.	.	.
C10_Value	N	N	bit data	dflt
CP0_Value	N	N	bit data	dflt
.	.	.	.	.
.	.	.	.	.
CP10_Value	N	N	bit data	dflt
Time_Stamp	N	N	TimeStamp	dflt

Table 3-2 EMS Conditions Table

3.2 Implementation

This section describes the implementation of the EMS design discussed in the previous section. The EMS was developed on the OS/2 2.1 platform, using Microsoft C Compiler version 6.0 and an expert system shell called Nexpert Object.

3.2.1 The Software Environment

3.2.1.1 OS/2 and Presentation Manager

OS/2 is an operating system which allows different applications, and multiple instances of the same application to run simultaneously, thus creating a multitasking environment. Furthermore, OS/2 supports multiple threads of execution within an application hence speeding up the performance of the executable. OS/2 also supports dynamic data exchange (DDE) of resources between two or more applications employing system semaphores to coordinate the data sharing.

The graphical user interface (GUI) of OS/2, the Presentation Manager (PM), provides a windowing interface. It equips the programmer with powerful graphical tools, and furnishes the end user with easy mouse interactions through icons, menus, and dialog boxes.

3.2.1.2 OS/2 Database Manager

The OS/2 Database Manager is a relational database management system (DBMS) which is fully integrated into the OS/2 environment. It provides an application programming interface (API), with which external programs can interact, and a Query

Manager, a graphical front-end interface which acts as a buffer between the user and the lower-level functionality of the database.

Operations and data relations in the DBMS are performed through statements of the Structured Query Language (SQL), which can be embedded in the high level language source code of the EMS.

3.2.1.3 Nexpert Object

Nexpert Object is a commercially available knowledge-based program that manages the expertise of a human expert. It consists of an inference engine and one or more knowledge bases created by an application to store domain knowledge in terms of objects, classes and properties.

The API of Nexpert Object consists of a set of routines that can be called from an external program. Using those routines, tasks such as starting the expert inference engine, finding the value of a property slot, or suggesting a hypothesis can be performed [Neuron Data, 1990]

3.2.2 Filter Module

The filter module receives raw data packets from the DLC module every minute. After smoothing the trends and providing level descriptions, it sends the processed data down the pipeline to the expert system for diagnosis. Linked lists of patients and trends coordinate the flow of data in the filter module as shown in figure 3.5.

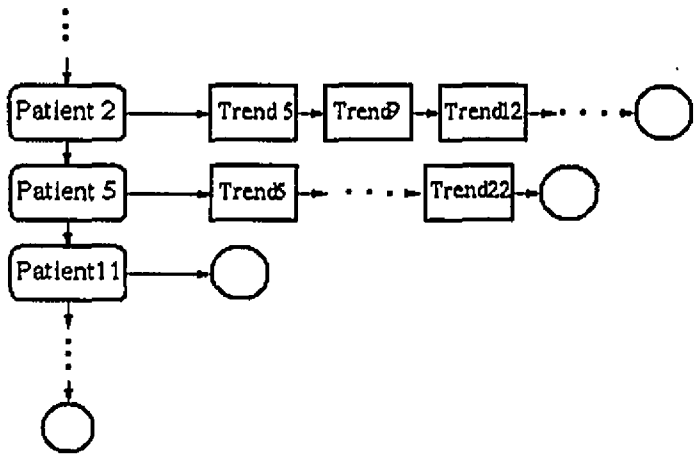

Figure 3-5 Patient and Trend Linked Lists in the Filter Module

As a patient is registered into the PDMS, a new node is created and added, in sequential order, to the linked list of patients. A new trend node is added to the trends linked list when new parameters arise. This node contains the different data structures essential for the rest of the filtering process.

The 13 and 17-point median filters are implemented with circular queues and linked lists, as shown in figure 3.6. Circular queues keep data points in the order that they appeared, while linked lists keep them sorted by increasing value.

When a new point is added to the end of the queue, the oldest point at the head of the queue is removed. The new data is also added at the appropriate location of the linked list. This insertion and deletion is performed in $O(n)$, where n is the length of the median filter.

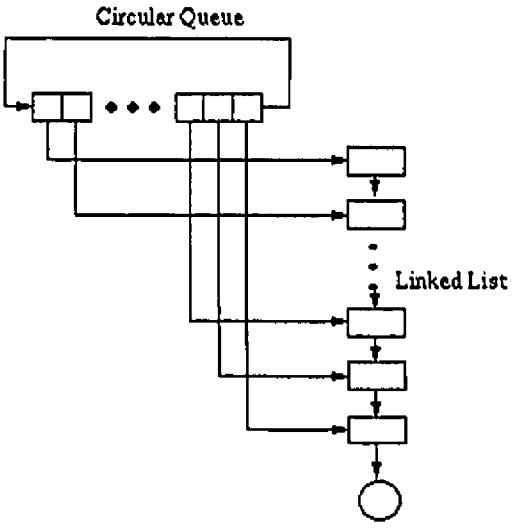

Figure 3-6 Median Filter Implementation

The FIR filter is implemented with a circular queue to store the data and an array that holds the filter coefficients, as illustrated in figure 3.7. Each data point in the circular queue of the FIR filter is obtained after passing the 17-point and the 13-point median filters. The net processed output value of the module is attained by adding the products of the FIR circular queue data points and the corresponding filter coefficients. Figure 3.8 shows the overall effect of the linearization filters on a sample heart rate trend.

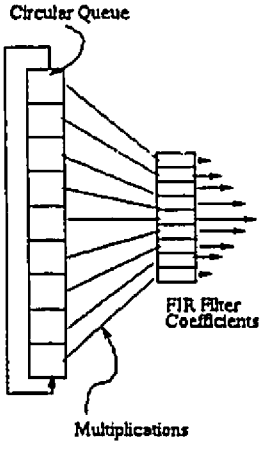

Figure 3-7 FIR Filter Implementation

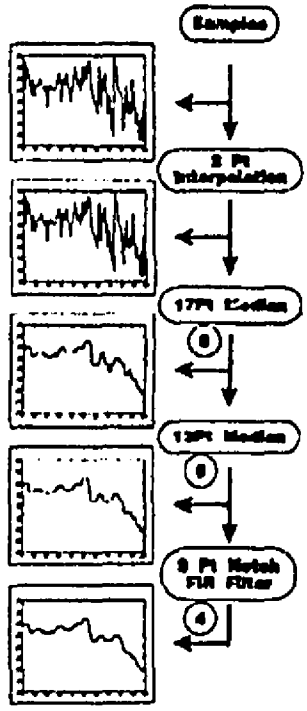

Figure 3-8 Effect of Linearization Filters on a Sample Trend

3.2.3 Rule Based Expert System

The domain of the expert system is modeled in terms of objects, classes and properties. Slots, which indicate specific properties of objects, store all the information needed by Nexpert to perform reasoning.

Objects in the knowledge base of the EMS can be divided into two groups: objects that hold the medical information for reasoning, and objects for controlling the inference strategy. Figure 3.9 shows a part of the class and object network of the expert system. The objects under *Actual* and *Predicted* classes hold the certainty indices, while *Raw* and *Processed* classes contain objects that hold prefiltered and filtered parameter

values, respectively. The objects under the *Cycle* class control the inference strategy and are used to manage the rules of the knowledge base.

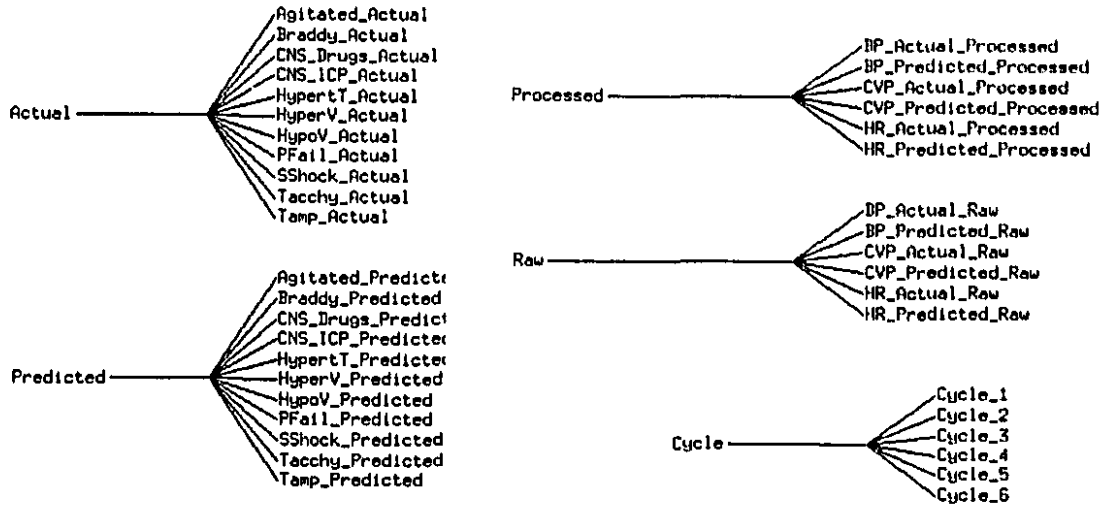

Figure 3-9 Part of the Object Network of the EMS

Any rule in Nexpert must be a list of implicitly ANDed conditions which produce a unique conclusion accompanied with many actions. Backward chaining is performed only with respect to the conclusion, not the actions of the rules.

The rules in the implementation of this expert system can be categorized into two sets. The first set is the one encoding table 3.1. One of the 27 rules is shown in figure 3.10. Each rule corresponds to a combination of parameter levels (HR, BP, CVP). Equation 3.2 to 3.7 are encoded in the right-hand-side of the production rule. Therefore the certainty index of the darkened entries in table 3.1 (positive diagnosis) will be increased by the fired rule, while those of white entries (negative diagnosis) will be decreased.

The second set of rules coordinate the four different types of parameters (actual-raw, actual-processed, predicted-raw, predicted-processed) and presents them to the first set of rules. Thus the 27 production rules can be applied on different types of parameters. Table 3.3 presents the sequence of Boolean slots that control the reasoning process.

```
@RULE= ES_223
        (@LHS=
                (Yes        (DiagnosExec))
                (=          ('HR_'\SysVar.Trend_Type\.Status_Level)    (0))
                (=          ('BP_'\SysVar.Trend_Type\.Status_Level)    (0))
                (>          ('CVP_'\SysVar.Trend_Type\.Status_Level)   (0))
        )
        (@HYPO=             ReachedDiagnosis)
        (@RHS=
                (Do         (MAX('HR_'\SysVar.Trend_Type\.Status_Level,\
        'BP_'\SysVar.Trend_Type\.Status_Level,'CVP_'\SysVar.Trend_Type\.Status_Level)*Sy\
        sVar.Weight_Type)   (Adj))
                (Do         ('HypoV_'\SysVar.Cond_Type\.Index-Adj*(1-SysVar.Weight_Found))
        ('HypoV_'\SysVar.Cond_Type\.Index))
                (Do         ('HyperV_'\SysVar.Cond_Type\.Index+Adj*(1-SysVar.Weight_Found))
        ('HypertT_'\SysVar.Cond_Type\.Index))
                (Do         ('SShock_'\SysVar.Cond_Type\.Index-Adj*(1-SysVar.Weight_Found))
                .
                .
        ('SShock_'\SysVar.Cond_Type\.Index))
                (Do         ('Agitated_'\SysVar.Cond_Type\.Index-Adj*(1-SysVar.Weight_Found))
        ('Agitated_'\SysVar.Cond_Type\.Index))
        )
)
```

Figure 3-10 Example of a Diagnostic Rule

Slots	Cycle0	Cycle1	Cycle2	Cycle3	Cycle4	Cycle5	Cycle6
New-Data-Done	T	F	F	F	F	F	F
Pat-Num-Done	F	F	T	--	--	--	--
Act-Proc-Done	F	--	F	T	--	--	--
Act-Raw-Done	F	--	--	F	T	--	--
Pred-Proc-Done	F	--	--	--	F	T	T
Pred-Raw-Done	F	--	--	--	--	F	T

Table 3-3 Slots for Controlling the Action of Reasoning

Cycle 0 is the initial state of the processing cycle. New-Data-Done slot is changed from True to False in cycle 1 as a new data set arrives. The four types of parameters are processed during cycles 2 to 5. After completing all computations, the inference engine is reset until a new data set arrives.

The expert system is implemented, along with the filter module, as a C program. The Nexpert API routines are also used to interact with the Nexpert Object shell. Figure

3.11 shows the architecture of the expert system. Based on the parameters submitted (raw and processed), the patient condition rules update the previous actual and predicted certainty indices.

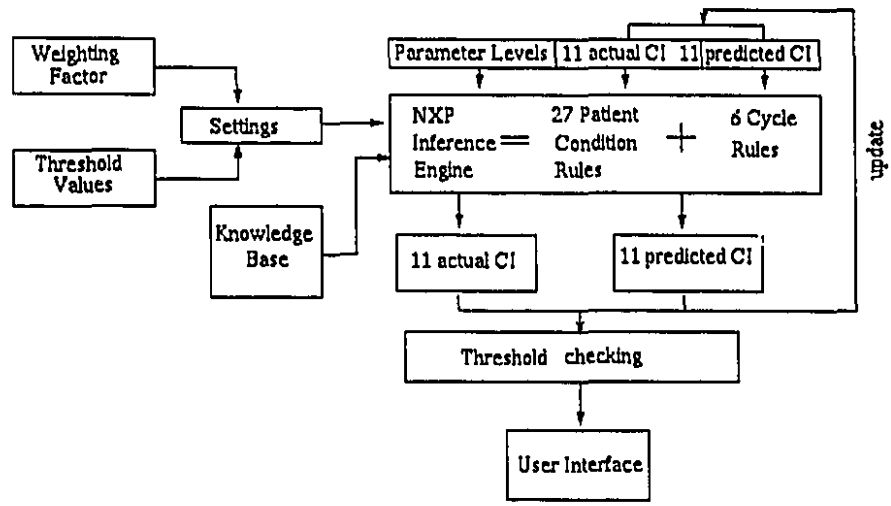

Figure 3-11 The Architecture of the Expert System

3.2.4 The User Interface

The user interface is implemented using the services of the OS/2 Presentation Manager APIs. Fig 3.12 shows the main window of the user interface which displays all the fourteen beds of the ICU simultaneously. The real-time status of each patient is updated and displayed every minute. A color coding scheme is used to indicate the patient's condition, with red signifying critical, yellow alarming, and green normal conditions. The bed color denotes the current condition, while the border of each bed represents the future predicted condition color. By selecting a bed number from the menu bar, or by clicking at a patient bed in the user interface, a window such as the one shown in figure 3.13 is displayed. This window presents the patients current and future conditions and their associated certainty indices. For both of these windows, it is

possible to change the text displayed from English to French or vice versa by simply selecting that option on the menu bar.

The Presentation Manager (PM) user interface is contained in an OS/2 session. However, Nexpert Object version 2.0 does not support PM services. As a result, two OS/2 sessions are required: one to run the PM and the user interface, while the other is needed to run the filter and the expert system program. Communications between the two sessions is achieved by the use of shared memory and OS/2 system semaphores. Separating the user interface and the expert system program in this way provides an added advantage. Since the only function of the user interface will be to display patient conditions, more than one user are can start an interface simultaneously to determine the results of the expert system program.

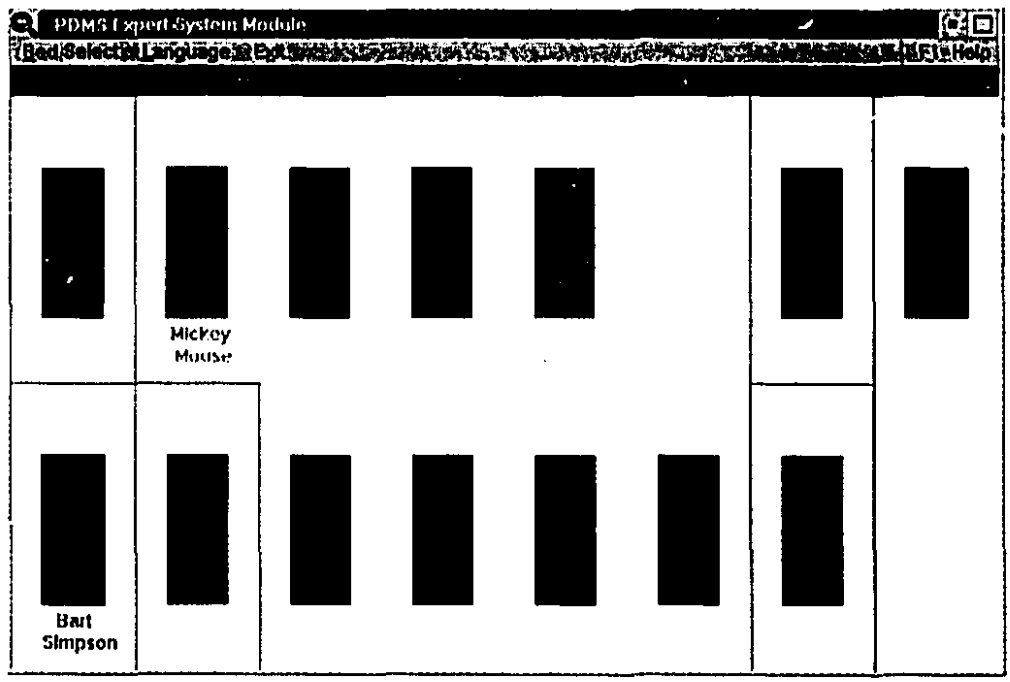

Figure 3-12 Main Window of the Expert Monitoring System

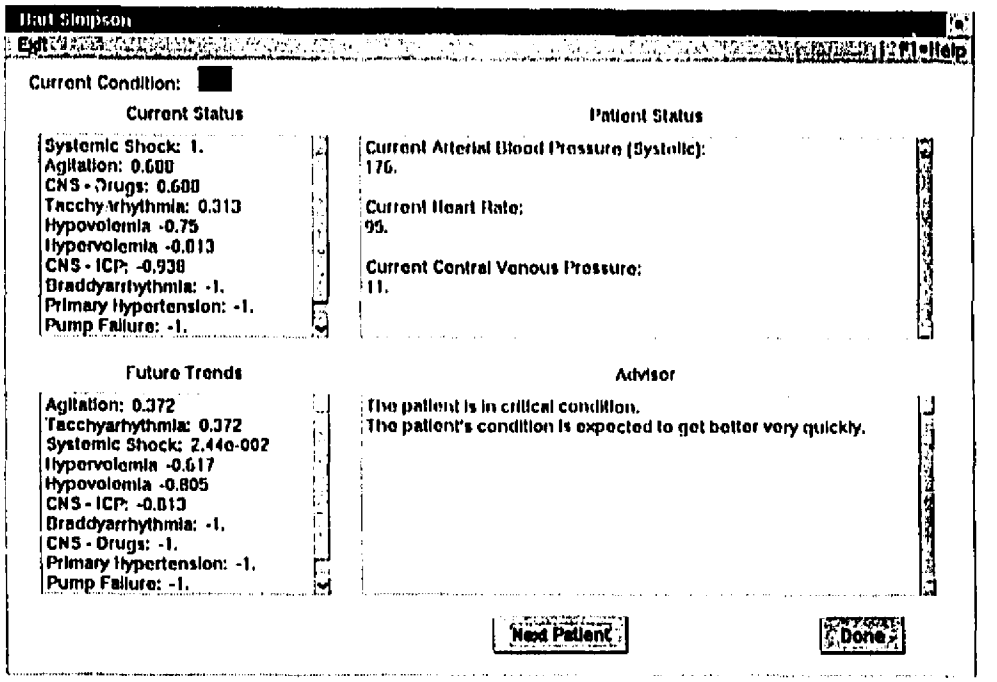

Figure 3-13 Patient Condition Level Interface

3.2.5 The Database Module

The database is implemented using the relational database manager services available from the OS/2 Extended Edition version 1.3 Database Manager. C code, with embedded SQL statements, is used to communicate with the Database Manager APIs. Figure 3.14 below shows a sample SQL statement that retrieves the data of each of the parameters into the corresponding host variables. A precompiler provided by the Database Manager preprocesses the embedded SQL statements before the entire code is compiled with Microsoft C compiler version 6.0

```
EXEC SQL
INSERT INTO EMS
(PATIENT_ID,DATE_OF_BIRTH, DATE_OF_READING, TIME_OF_READING,
HR, CVP, BP, OVERALL_A)
Values(:patid, :dateofb, :dateofr, :timeofr, :hrate, :cvpressure, :bpressure,
:cond);
```

Figure 3-14 SQL Statement for Transferring Data into Host Variables

In the current implementation of the DLC module, parameter data acquired are stored in memory-resident queues every minute (see section 2.4.1). However, saving into the database is not done at that rate. Rather, the data is transferred in blocks to the disk at periodic time intervals. This is done to avoid excessive disk access which may slow the system down. Furthermore, using such a periodic transfer gives an added advantage: data blocking offers the opportunity for other programs to interact with the database because it is not locked.

The frequency and the length of time in which a database would be locked, and consequently the inability of other programs to perform operations on it, is dependent on the chosen value for periodic transfer. This interval is currently set at 5 minutes in the EMS module.

Chapter 4 Evaluation of System Performance

Software evaluation aims at assuring the compliance of a program with its requirements, which captures the needs of the end-user. This concept remains fully applicable when the target software is an expert system, although expert system particularities demand specific evaluation criteria. The system's efficiency and the quality of its decisions and assessments are areas of focus in the evaluation of the EMS. This chapter begins the evaluation process with the subsystem validation of the filter module and the verification of the knowledge base, each done separately. The advantage of testing in this way is that the error is localized, hence error-detection is easier.

In the next step, the integrated EMS with all its submodules is evaluated for its performance. At this stage, testing is performed using a simulator. In the final phase of evaluation, field tests and evaluation are carried out on the quality of decision of the EMS. Two approaches of visualizing the results are examined before concluding with future work.

4.1 Subsystem Evaluation and Validation.

4.1.1 Sensitivity Analysis on Filter Module

The sensitivity analysis of the filter module is divided into three steps:

1. Boundary value analysis;
2. Time delay analysis of the filters;
3. Sensitivity towards noise input.

The inputs of the filter module are the three monitored physiological parameters, heart rate (HR), blood pressure (BP) and central venous pressure (CVP). The boundary value of the filter was tested by varying the input values from -100 to +500. However, the filter is programmed to treat negative values as invalid, thus the lower bound of the input is zero. Moreover, any BP or HR value above 400 and any CVP above 100 are also treated as invalid and are rejected by the filter module. The test showed that the filter complied with the programmed restrictions. Therefore upper bounds and lower bounds exist and are clearly defined for each of the input parameters.

In the next step of the analysis, the delay of the filter module is examined. As was explained in Chapter 3, different parts of the filter module result in the delays shown in Table 4.1.

Unit	Delay Points	Delay Time
2-Point Interpolation	1	1/2 minute
17-Point Median Filter	8	4 minutes
13-Point Median Filter	6	3 minutes
9-Point Median Filter	4	2 minutes

Table 4-1 Delay of the Filter Module

From the table above, the overall delay time is seen to be 9.5 minutes. To verify this, two hours of a simple input parameter data were input to the filter module, and the delay was determined by observing the time shift in the output data. Figure 4.1. presents these results. As indicated in the figure, the outcome is consistent with the design.

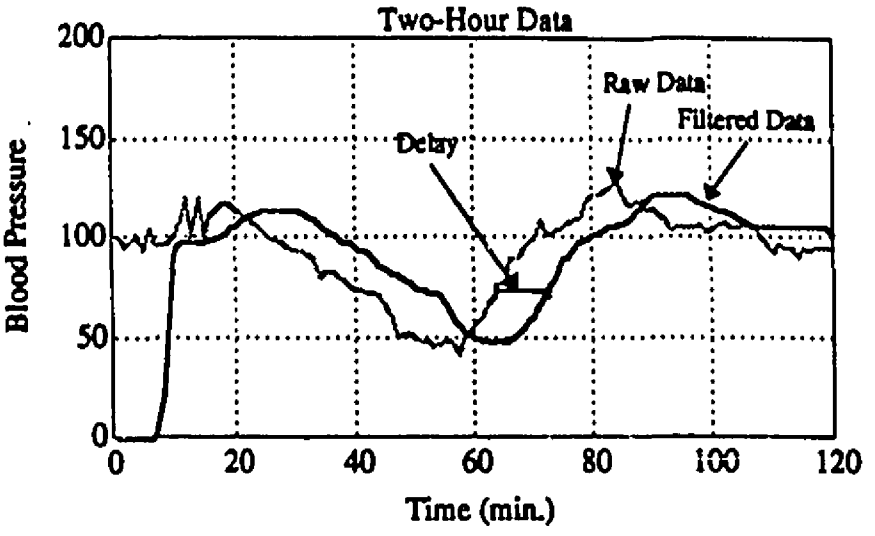

Figure 4-1 Delay of the Filter Module

In the last step of the sensitivity analysis, the behavior of the filter with respect to input noise was investigated. The test consisted of generating square impulses ranging in width between one and seven minutes, and using them as input to the filter module. The filter treats any impulse of a width three minutes or less as noise data and filters it out. This is shown in Figures 4.2 and 4.3.

4.1.2 Knowledge Base Verification

Comprehensive verification of knowledge base must be carried out before a system can be effectively used. Evidence from the computer industry suggests that without such verification, knowledge base will not be upgradable and the expert system implementing them will not be safe and reliable for field use [Xue, 1993].

Verification on the EMS is done via consistency and completeness tests of the knowledge base.

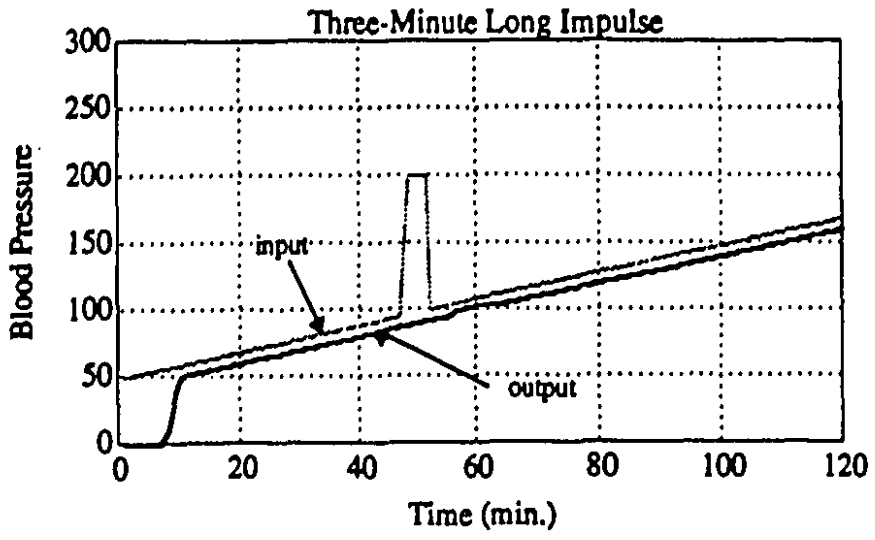

Figure 4-2 Impulse Response with 3 minute Square Impulse

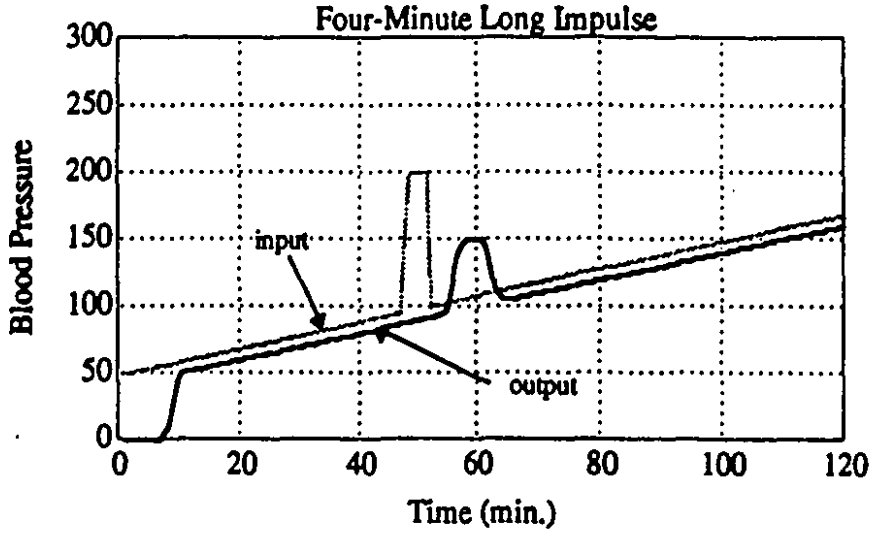

Figure 4-3 Impulse Response with 4 minute Square Impulse

4.1.2.1 Method

As was mentioned in Chapter 1, consistency checking is achieved by testing for the following:

1. Redundant rules: rules which succeed in the same situation and have the same conclusion.
2. Conflicting rules: rules which succeed in the same situation but with conflicting conclusions.
3. Subsumed rules: two rules having the same conclusions but one contains additional constraints or the situation in which it will succeed.
4. Rules with unnecessary if conditions: rules which have the same conclusions, but the if condition in one rule is in conflict with one if condition in another rule, and all other if conditions in the two rules and equivalent.
5. Circular rules: the chaining performed on a set of circular rules forms an endless circle.

Completeness test is carried out by inspecting the objects and slots of the knowledge base, and eliminating the following situations:

1. Unreferenced slot value: a slot value which is not covered by any value's if conditions.
2. Illegal slot value: a slot value does not belong to the set of legal values.
3. Terminated goal: goal that terminates the reasoning process with an unsatisfied result.
4. Terminated if conditions: conditions causing the terminated goal.

4.1.2.2. Results

The knowledge base of the EMS consists of two sets of production rules. As previously discussed in Chapter 3, the first set encodes the patient conditions, and the second set manages the action of the first set.

The verification process on these two sets was done with the debugging aid of the Nexpert Object expert system shell with its built-in rule syntax checking, object usage and rule consistency checks, along with its rule and object network graphs. Tables 4.2 and 4.3 summarize the obtained results.

Although circular rules should be avoided, those flagged in Table 4.3 are there by design. They are used for iterative looping continuous operation of the twenty-seven patient condition rules. Thus the results show that the knowledge base of the EMS are consistent and complete.

Consistency of Knowledge Base	Results
Redundant Rules	Nil
Conflicting Rules	Nil
Subsumed Rules	Nil
Unnecessary IF Conditions	Nil
Circular Rules	Nil
Completeness of Knowledge Base	
Unreferenced Slot Values	Nil
Illegal Slots	Nil
Terminated Goal	Nil
Terminated IF Conditions	Nil

Table 4-2 Verification of the First Set of Rules

Consistency of Knowledge Base	Results
Redundant Rules	Nil
Conflicting Rules	Nil
Subsumed Rules	Nil
Unnecessary IF Conditions	Nil
Circular Rules	6
Completeness of Knowledge Base	
Unreferenced Slot Values	Nil
Illegal Slots	Nil
Terminated Goal	Nil
Terminated IF Conditions	Nil

Table 4-3 Verification of the Second Set of Rules

4.2 System Validation By Simulation

4.2.1 The Simulator

Placing the EMS in a critical care environment such as the ICU not only requires extensive verification and validation to ensure its proper functioning, but also proper training of the medical staff. A patient data simulator was used to serve this dual purpose [Fumai et. al., 1990]. The simulator mimics the real setting of the ICU by synthesizing patient data and transmitting it to the host computer through a serial line, thus performing the functionality of the CareNet network and the Careport interface described in Chapter 2. The simulator runs on a separate machine, an IBM PS/2 Model 80, with an 80386 CPU running 16 MHz and 8 MB RAM. The data generated is transmitted through an RS-232C serial link to the PDMS computer.

4.2.2 System Performance

The EMS receives physiological data and generates results every minute. To demonstrate the capabilities of the EMS to carry out such a time-critical task, a performance test was carried out using the simulator described above. The goal of the test was to determine the time needed by the EMS to interpret vital signs and generate all the ICU patient results. The response time was measured by inserting statements in the C code of the EMS that send the OS/2 system clock before and after each EMS cycle. The difference between those two results in the time response. The test began by admitting one patient. This was repeated fourteen times, incrementing the number of admitted patients by one with every iteration. The results are shown in Figure 4.4.

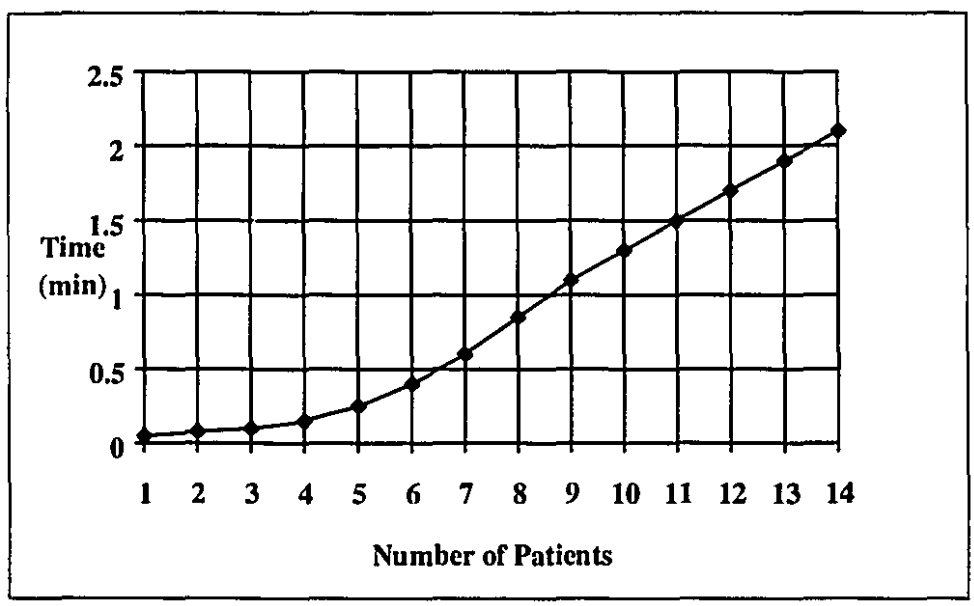

Figure 4-4 Time to Interpret Patient Conditions

The results demonstrate that the EMS fails to meet the requirements after eight patients have been admitted. Under such circumstances, the response time exceeds the one-minute acquisition rate. The main reason for the slow execution speed is that the knowledge base of the EMS developed by the Nexpert Object had to be stored in text format. Compiled versions of knowledge bases, also available from Nexpert Object

under the UNIX and Windows environments, demonstrates a speed-up factor of 10. Unfortunately, Nexpert (Version 2.0) does not support a compiled version of knowledge bases under the OS/2 environment. Thus its compilation during runtime takes a major portion of the CPU time.

Three solutions present themselves at this point:

1. Using compiled version of the knowledge base if a future release of Nexpert Object supports it.
2. Using a faster machine, preferably a Pentium as the PDMS host computer. This is estimated to enhance the performance, at least by a factor of four.
3. Partitioning the EMS to run on one machine and the rest of the PDMS on another machine.

The reason for the last solution is that the Data Link Controller module acquires large volumes of data from the Careport unit every two seconds, which presents heavy CPU loading. Separating the EMS and the DLC reduces the competition for CPU time and improves the overall performance.

4.2.3 Database Considerations

Further complications to the performance of the EMS arise from the fact that parameter data, including the three vital signs and the eleven actual and predicted condition values have to be updated in the database every minute. Figure 4.5 displays the time response of the database updates. The numbers in the graph were obtained by running a test in which patients were added one at a time to an empty database and calculating the response time using the OS/2 system clock as explained earlier. However, a general degradation in these values is expected as the database becomes larger.

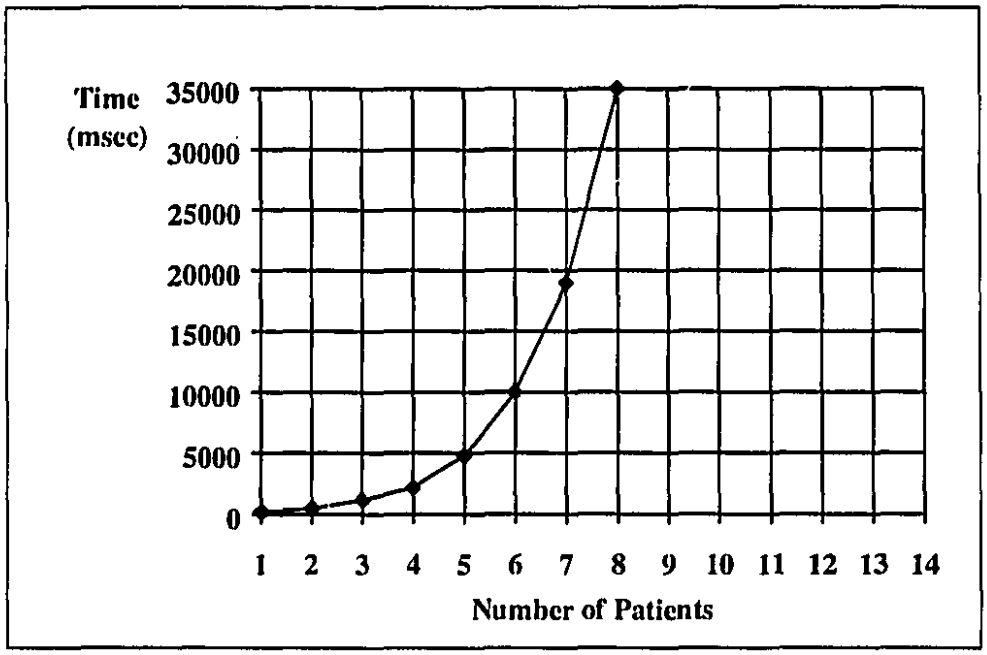

Figure 4-5 Time of Database updates

The figure shows that the performance is satisfactory at the beginning but severely deteriorates as the number of patients increase.

The performance of the system is enhanced by increasing the buffer size of the database and updating it in cycles longer than one minute as was mentioned in section 3.2.5. An update cycle of 5 minutes, currently used in the EMS, showed an overall improvement of 48% to the numbers shown in figure 4.5. This number is acceptable at this stage but may require further investigation, particularly when running a regular back up process on the database.

Regular back up is important for evaluation purposes, however the backup utility in OS/2's Database Manager connects to the database in exclusive mode, that is once the utility starts it prevents other applications from accessing the database until the backup is complete. This adds to the inherent complexity expected with the connection of the database to the time critical EMS and warrants additional analysis.

4.3 Field evaluation

Wyatt and Spiegelhalter indicate that expert system testing could not be limited to the laboratory, and it could not be carried out only by the developers of the system. Instead, testing must sample real situations [Wyatt and Spiegelhalter, 1990].

The EMS was transported to the Montreal Children's Hospital to conduct field tests in order to evaluate the effectiveness of its decisions on real patients, and to locate any incurring weaknesses. Two schemes of visualizing the results were explored and are described next.

4.3.1 Visualization

Visualization is now recognized as a fundamental element in scientific computing. As high-performance computing allows users to solve ever-larger problems, and as measurement technology yields ever more data to be studied, the ability to analyze and understand this data relies more and more on visual processing [Brodlie, 1995]. This is certainly applicable to real-time expert systems and the EMS benefits immensely from having visual aids to complement its application and evaluate its performance.

The two visualization schemes used were: the conventional 2D plot, and a more elucidating 3D display.

4.3.1.1 Two-Dimensional Display and Analysis

In the first strategy used to investigate the results, the output data generated by the EMS are plotted using a minutes temporal time index on 2D graphs. An example is shown in figure 4.6.

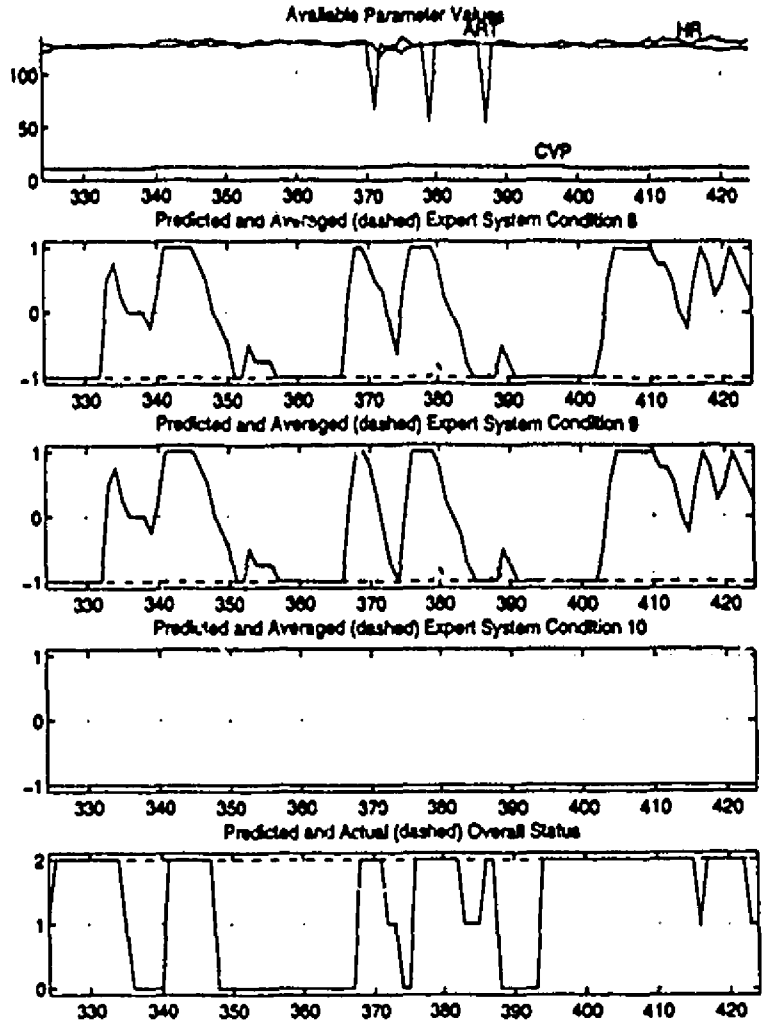

Figure 4-6 Two Dimensional Evaluation Plots

Along with the monitored parameters, the status assessed by the EMS for each of the eleven patient conditions is presented. Moser et. al. maintain that important information is gained from viewing physiological patient parameters simultaneously [Moser, et. al., 1990]. However, to do so using the 2D plots and to assess the performance of the EMS, doctors have to sift through several graphs and give interpretations on different locations of the graph. This task is tedious and time-consuming. A more efficient way of viewing the information is by taking advantage of the fact that three physiological parameters are being monitored, thus using them to obtain three-dimensional displays.

4.3.1.2 Three Dimensional Display and Analysis

The data representation in the 3D visualization approach takes the form of a point in space with its position being determined by the three monitored physiological variables. In this representation, the time is not displayed graphically. Instead, each point represents a distinct time. In other words, the three parameters are measured at the same time for each point. The location of the points as well as their shape and color are used to distinguish the varying circumstances generated by the EMS. On color displays, green Os denote stable conditions while red X's indicate critical conditions [Abu-Shihab et al., 1995]. Figure 4.7 shows a sample 3D plot that displays the result of the data generated after running the EMS for a period of three hours (180 points are produced each corresponding to one minute).

By analyzing figure 4.7, two advantages of this method of visual display are noticeable. First, the three dimensional aspect of the spatial representation can be used to identify clustering properties of points and conditions that may be associated with them. In this way, spatial locations for the eleven different diseases may be recognized and studied. Avanzolini et al. used similar graphs to study and evaluate the diversity between two distributions [Avanzolini et al., 1991]. The same idea may be used here to measure class separability between normal and critical conditions.

The second advantage of the 3D display lies in the fact that a boundary that includes all the points of one kind, such as the stable conditions, can be deduced. This is shown in figure 4.6 as a three dimensional bounding box that represents the stable region of the patient. The stable bounding box configuration varies according to the patient's age. Any point outside the box should be considered as critical and be labeled as X by the EMS. Therefore this method provides an easy way of assessing the quality of the decision of the EMS. Results obtained from field tests and analyzed using three dimensional plots are described next.

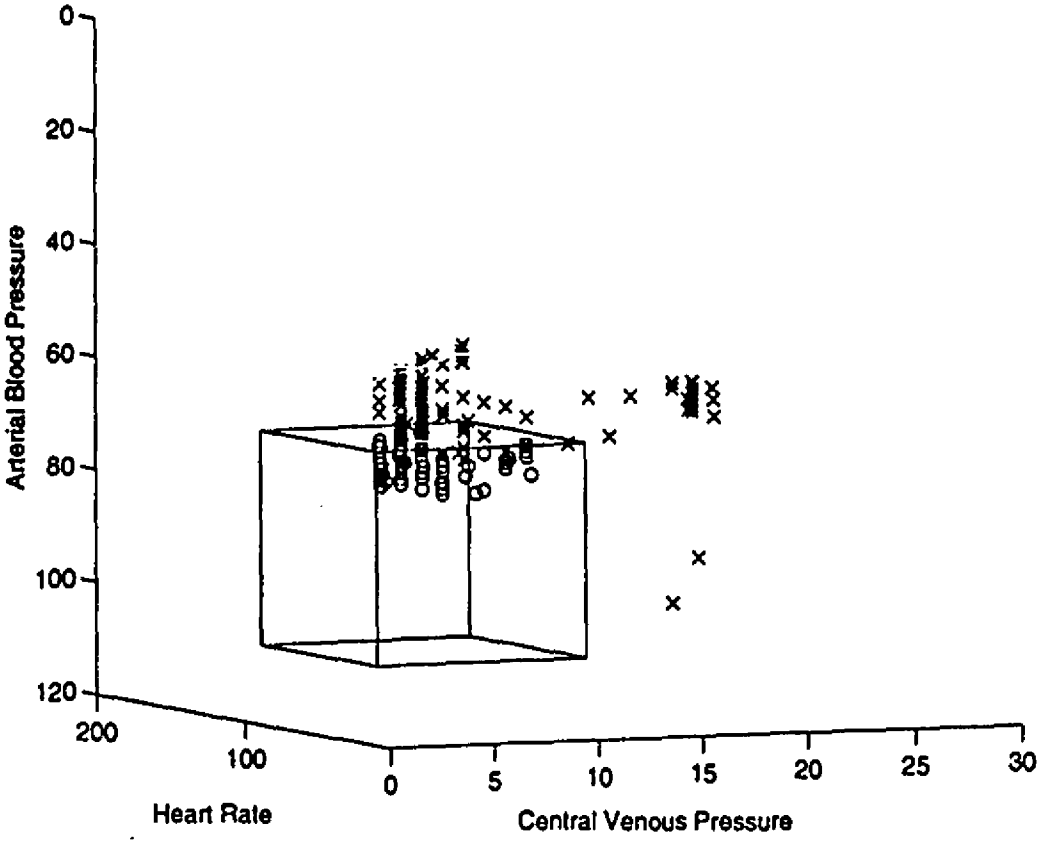

Figure 4-7 Three Dimensional Evaluation Plots

4.3.2 Results and Discussion

The EMS was allowed to run at the Montreal Children's Hospital for a period of three days. The data generated by the EMS during that period was analyzed applying the 3D visual displays. Using the bounding box criteria mentioned above, along with standard values obtained from the physicians at the hospital, it was possible to identify errors and to classify them as *false positive* or *false negative*. The standard values representing the ideal stable region of a patient are shown in table 4.4. The subscripts

'l' and 'h' in the header of the table represent low and high respectively. Values between these highs and lows are expected to be stable.

Age	BP_l	BP_h	HR_l	HR_h	CVP_l	CVP_h
0-14 days	53	91	70	190	5	15
15-29 days	60	98	70	190	5	15
30-44 days	66	104	80	160	5	15
45-59 days	70	108	80	160	5	15
60 days-1year	72	110	80	160	5	15
1-2 years	73	111	80	160	5	15
2-4 years	75	113	80	130	5	15
4-6 years	77	115	80	120	5	15
6-8 years	79	117	75	115	5	15
8-10 years	84	122	70	110	5	15
10-12 years	88	126	70	110	5	15

Table 4-4 Standard Values for Stable Conditions

Errors may be detected by matching, for each case C_i, the set of hypothesis H_i that the expert system assigns to C_i against the correct set of hypothesis H_i* [Meseguer, 1995]. An hypothesis h is classified in:

1. *False positive*, if $h \in H_i$ and $h \notin H_i$*
2. *False negative*, if $h \notin H_i$ and $he \in H_i$*

In the case of the EMS, if a point is diagnosed as critical but lies within the region of the bounding box, the point is considered as *false positive*. On the other hand, if a point is labeled as normal but lies outside the region of the boundary box, then it is considered as false negative. Clearly False negatives pose a greater threat to the

functionality of the EMS than false positives. This is true because a patient in a critical condition may not get the attention required.

Four Megabytes of data were acquired after monitoring four patients for a period of three days. After performing the 3D analysis, the results shown in table 4.5 were obtained. The errors are expressed in terms of percentages of the total number of points found under each condition, i.e. for false positives the number of Xs inside the bounding box over the total number of Xs obtained, and for false negatives the number of Os outside the bounding box over the total number of Os obtained. Points shown in table 4.5 were gathered at one minute time intervals.

	correct Os	incorrect Xs	incorrect Os	correct Xs	False Positive	False Negative
Patient 1	470	155	135	236	39.25%	22.3%
Patient 2	337	41	74	88	31.7%	18.1%
Patient 3	422	137	89	252	35.2%	17.4%
Patient 4	310	109	71	210	35.2%	18.6%

Table 4-5 Results of Expert System

The EMS condition classifications for the four patients during the monitored period are shown in table 4.6.

For further investigation of the results obtained in table 4.5, 2D temporal plots were used to find the locations in time of the false positives and false negatives. Figures 4.8 to 4.11 show the 2D plots of the HR, BP, CVP, current and predicted conditions assessed by the EMS, along with the false positive and false negative plots for each of the patients in table 4.5.

	Actual Classification	Predicted Classification
Patient 1	51% Bradycardia 49% Pump Failure	24% Bradycardia 70% Pump Failure
Patient 2	46% Pump Failure 41% Bradycardia	46% Pump Failure 33% Bradycardia 15% Hypovolemia
Patient 3	47% Tamponnade 42% Pump Failure 11% Bradycardia	49% Tamponnade 48% Pump Failure
Patient 4	69.5% CNS-ICP 26% Bradycardia	75% CNS-ICP 19.7% Bradycardia

Table 4-6 EMS Classifications of the Patient Conditions

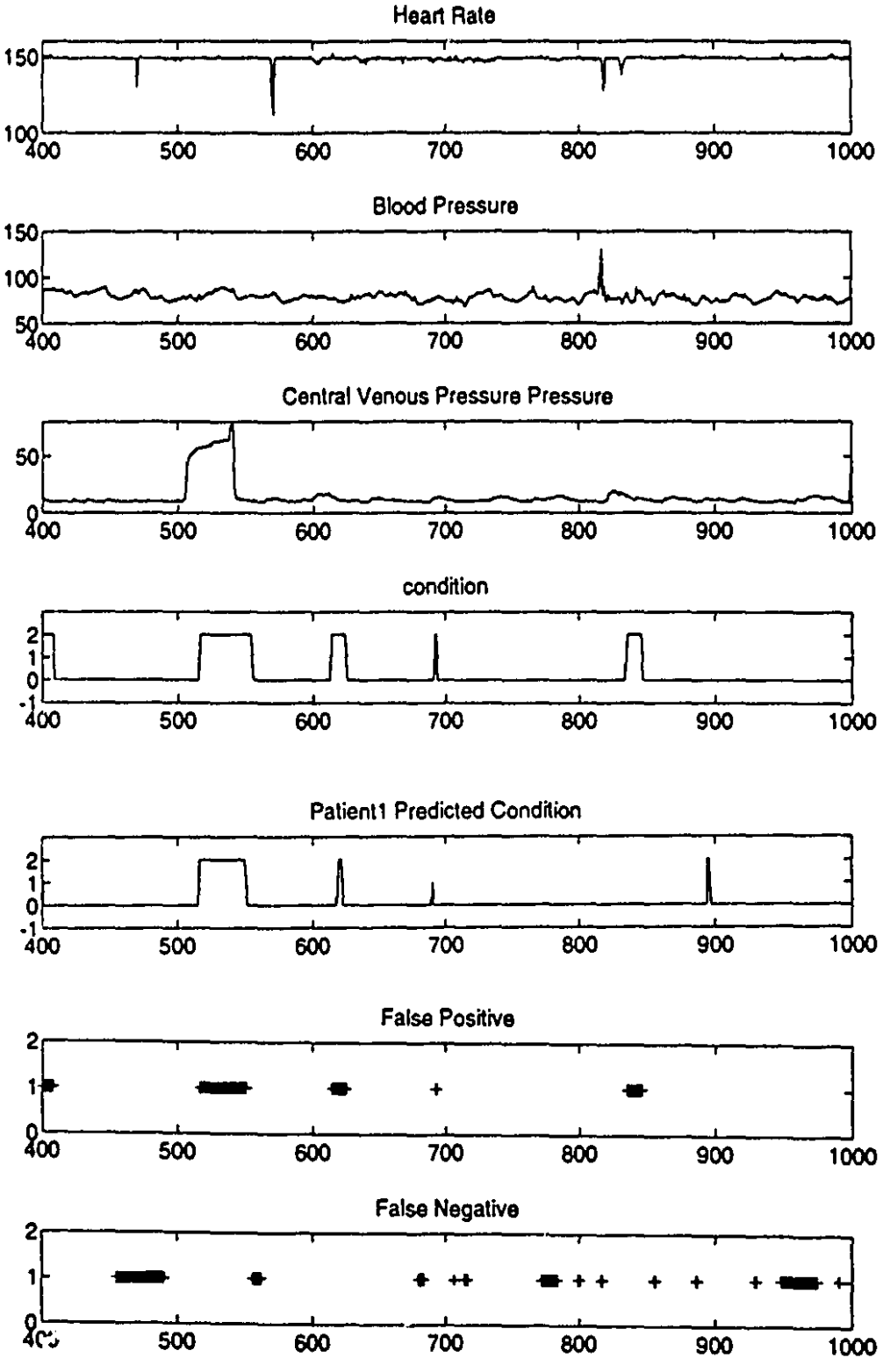

Figure 4-8 2D Time Plots of Patient 1

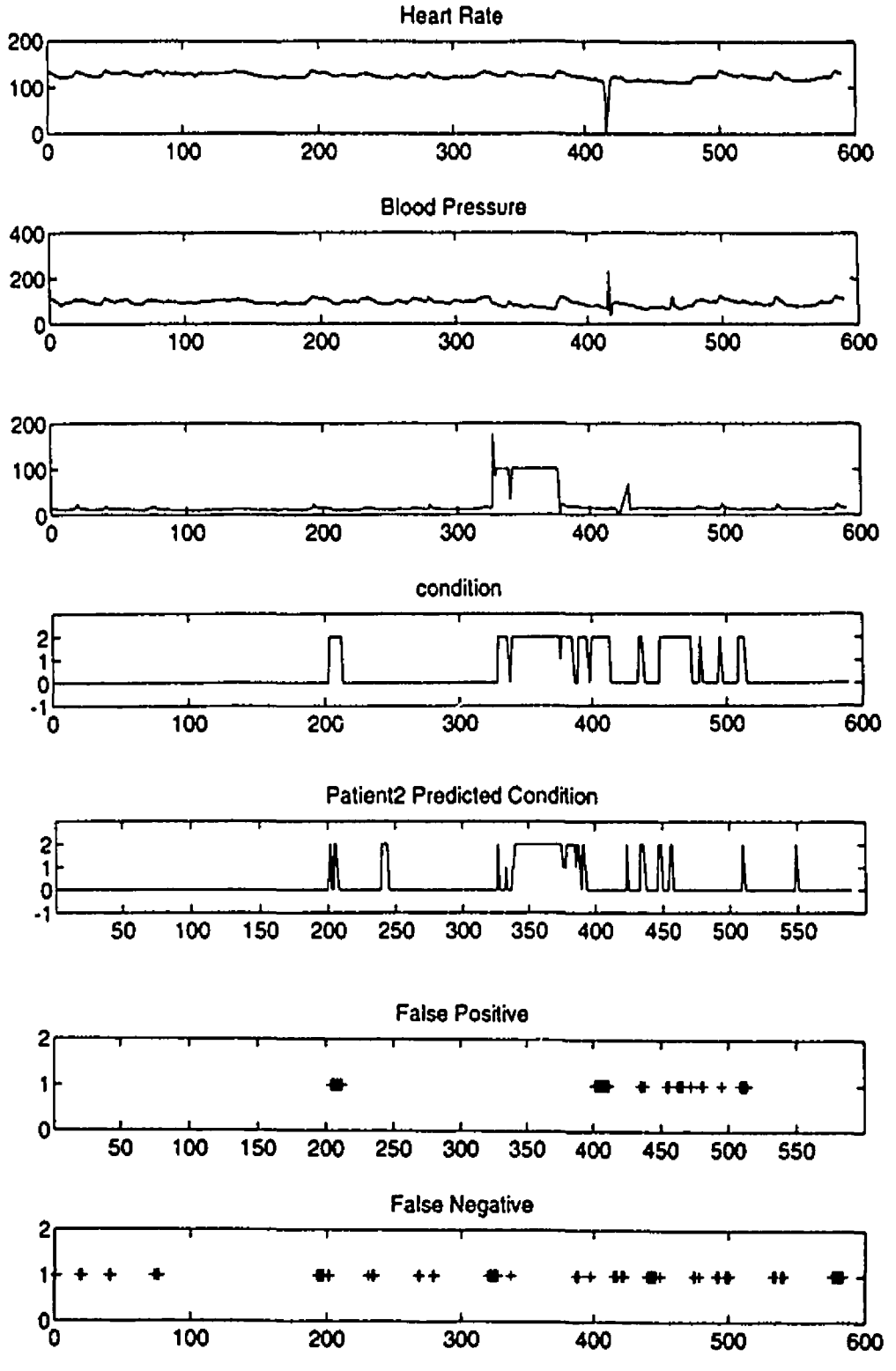

Figure 4-9 2D Time Plots of Patient 2

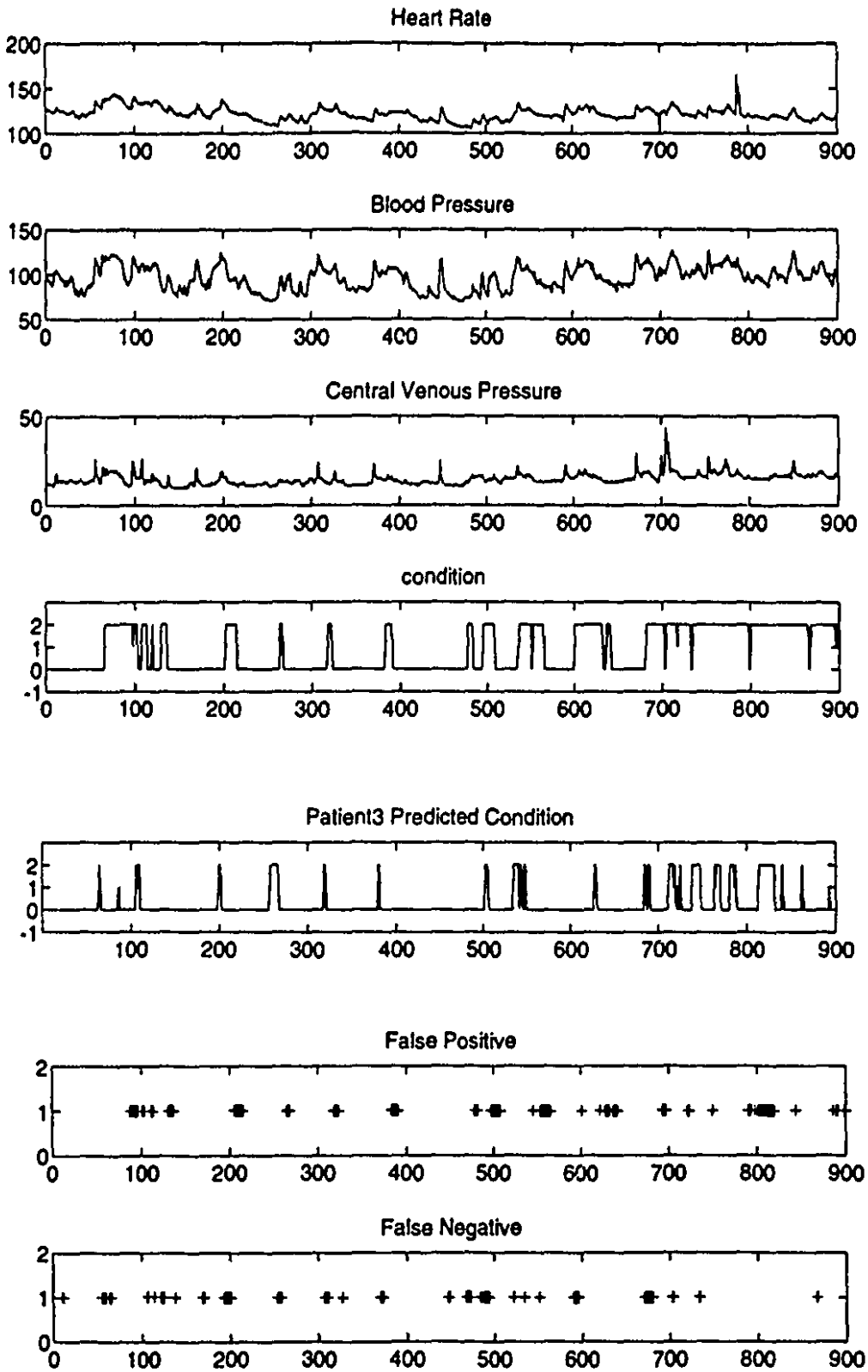

Figure 4-10 2D Time Plots of Patient 3

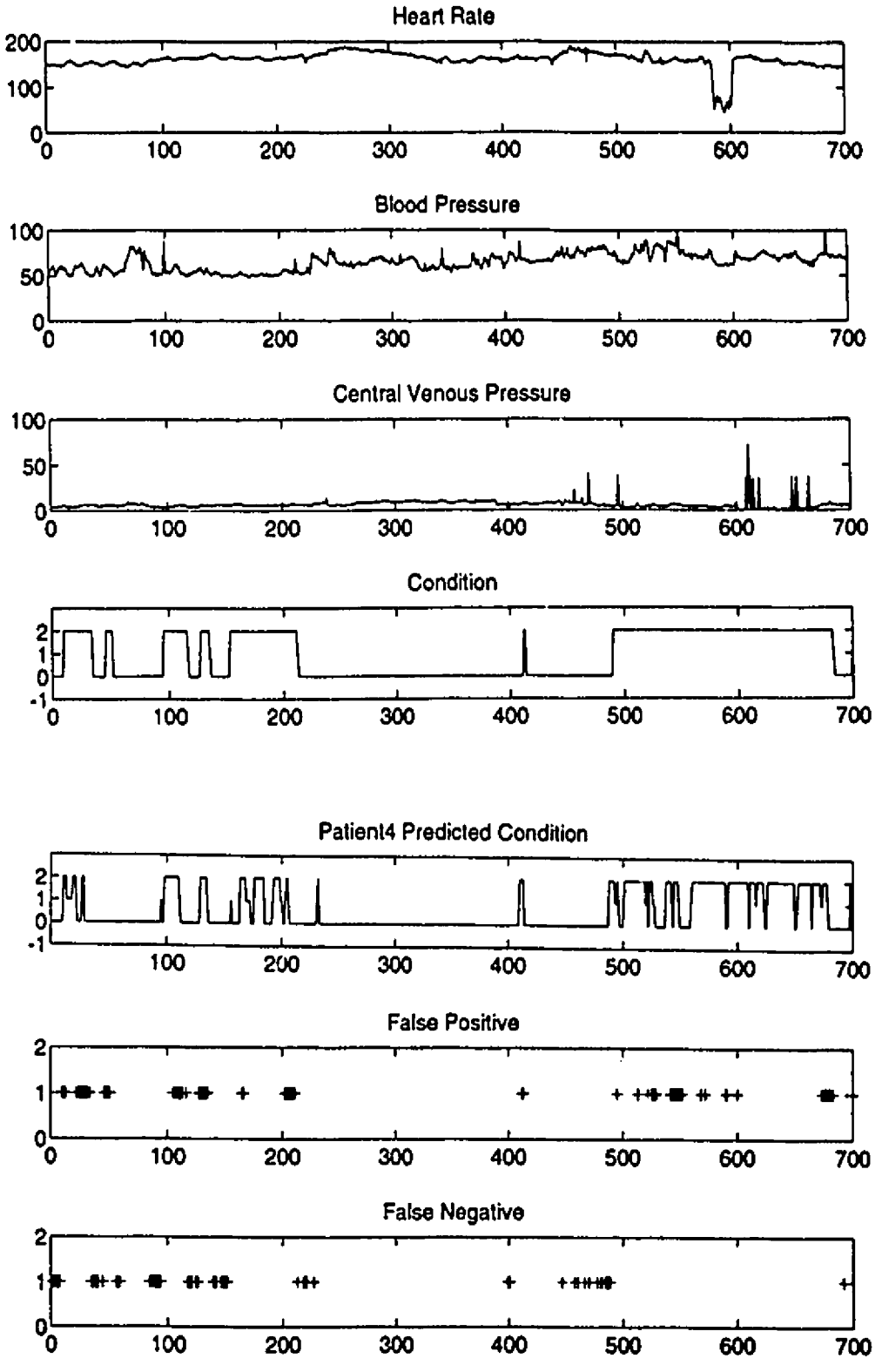

Figure 4-11 2D Time Plots of Patient 4

Upon inspection of the 2D plots in the figures above, it is apparent that the false positives and false negatives mostly fall in the regions where a change in the patient's condition took place, with false positives occurring in the neighborhood of the critical conditions and false negatives in the neighborhood of the normal conditions. This is expected because of the inevitable latencies of the EMS due to the time taken for the weights to track the current condition. This is further supported by the fact that a small number of errors occurred in regions of constant conditions.

Scrutinizing the classification accuracy of the EMS where false positives and false negatives occur produced results similar to those shown in table 4.6. Thus, the EMS produced correct condition classifications in spite of these errors.

Figures 4.8 to 4.11 also aid in demonstrating the prediction capability of the EMS. A careful scrutiny of the plots showed that the predicted condition occur slightly ahead of time of the actual conditions. However, some erratic situations are apparent as well. These warrant additional analysis in the future.

Finally, as a last step towards the evaluation process, the results of table 4.6 were discussed with the health care professionals at the Montreal Children's Hospital. They were of the opinion that the statistics produced by the EMS agreed with the actual medical interpretations observed in the intensive care unit. However, further examination was recommended.

4.4 Future Work

The knowledge base of the EMS is currently undergoing further evaluation and analysis. Refinement of the knowledge base rules is necessary. The success of the EMS is contingent on defining a correct set of production rules that lead to acceptable results.

The present EMS monitors the patient's blood pressure, heart rate, and central venous pressure. Although the combination of these three parameters provides a good description of the patient status, a future version of the EMS should support monitoring other physiological parameters as well as HR, BP, and CVP. Under such circumstances, different production rules will be needed for different sets of monitored variables. One way of achieving such a dynamic knowledge base would be by enabling the EMS to interrogate the database module for the additional desired parameters. In this way, new rules may be stored in the database which easily retrieve the appropriate set of physiological parameters.

Other grounds for improvement include the integration of an on-line, interactive three dimensional visual display. Currently, the displays are obtained using MATLAB under the UNIX environment. However, it is important that these displays be realized on the OS/2 platform and linked directly to the EMS and the database modules. This would allow real time viewing of the data as they are generated. A Presentation Manager user interface for the 3D displays should be incorporated that provides user input and allows inquiries about specific variables, points, or locations. The interface should also be capable of manipulating the data to furnish the user with different views from different angles. This work is currently in progress

Chapter 5 Conclusion

Over the past 25 years, there has been a great deal of research exploring how the computer might assist the clinician in areas of patient care. The ability of the computer to help assimilate the vast amount of information facing physicians every day has attracted considerable interest. Researchers have addressed these issues both in very practical ways and also from quite theoretical perspectives.

In this thesis, the development and evaluation of an Expert Monitoring System (EMS) was discussed. A literature review was presented that identified the need of data management and computerization in the ICU. The survey also stressed the necessity of formal verification and evaluation on expert systems generally and medical expert system specifically. Several methodologies of evaluation and verification used on some major existing medical expert systems were highlighted.

A description of the hardware and software architectures of the Patient Data Management System (PDMS) under development at the Montreal Children's Hospital was presented. The functionality of each of the submodules that form the PDMS was outlined.

The design, implementation, and evaluation of the EMS was described in detail. The results of the evaluation process were presented and discussed. two different schemes of viewing the field test results were analyzed and illustrated. Finally, future work on the EMS and recommendations for its improvement were outlined.

References

[Abu-Shihab et al., 1995] O. Abu-Shihab, V. Zia, F Al-Refaee, H. Leijonhufvud, A. Malowany, F. Carnavale, and R. Gottesman, "3D Visualization for Evaluating a Monitoring System in an ICU". *Proceedings of the IEEE International Conference on Engineering in Medicine and Biology*, p.488-489, 1995.

[Andreoli, 1985] K. Andreoli, "An Overview of How Computer Applications in Nursing Service Affect Patient Care". *Nursing Outlook*, vol. 33 (1), p.16-21, 1985.

[Autio et al., 1991] K. Autio, A. Kari, and H. Tikka, "Integration of Knowledge-Based System and Database for Identification of Disturbances in Fluid and Electrolyte Balance". *Computer Methods and Programs in Biomedicine*, vol. 34, p.201-209, 1991.

[Avanzolini et al., 1991] G. Avanzolini, P. Barbini, G. Gnudi, and A. Grossi, "Cluster Analysis of Clinical Data Measured in the Surgical Intensive Care Unit". *Computer Methods and Programs in Biomedicine*, vol. 35 (5), p.157-170, 1991.

[Ben-Basset et al., 1980] M Ben-Basset, R. Carlson, u. Puri, and M. Latif, "Pattern-based Interactive Diagnosis of Multiple Disorders: The MEDAS System". *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-2*, p.148-160, 1980.

[Boehm et al., 1978] B. Boehm, J.R. Brown, h. Kasper, M. Lipow, G.J. McLeod, and M.J. Merrit, *Characteristics of Software Quality*. North-Holland, New York, 1978.

[Bowes and Wilson, 1994] C.L. Bowes and A.J. Wilson, "Information Management Systems for Intensive Care". *Computer Methods and Programs in Biomedicine*, vol. 44, p. 31-35, 1994.

[Brodlie, 1995] K. Brodlie, "Scientific Visualization - Post, Present and Future". *Nuclear Instruments and Methods in Physics Research*, A354, p104-111, 1995.

[Brown et al., 1994] D.E. Brown, P.J. Shaw, S. Vittone, and K. Weise, "A Comparison of Inductive Modeling Techniques for Pediatric Decision Making". *IEEE International Conference on Systems, Man and Cybernetics*, p.919-924, 1994.

[Burkes, 1991] M. Burkes, "Identifying and Relating Nurses Attitude Toward Computer Use". *Computers in Nursing*, vol. 9 (5), p.190-198, 1991.

[Chang et al., 1990] C.L. Chang, J.B. Lombs, and A. Stachowitz, "A Report on the Expert System Validation Association (EVA)". *Expert Systems With Applications*, vol. 1, p. 217-230, 1990.

[Clayton and Hripcsack, 1995] P.D. Clayton and G. Hripcsack, "Decision Support in Health care". *International Journal of Biomedical Computing*, vol. 39, P.59-65, 1995.

[Collet, 1992] C. Collet and A. Malowany, "An Expert Monitoring System for a Pediatric Intensive Care Unit". CEFRIO Report, McGill University, Montreal, Canada, February 1992.

[Dan and Dudek, 1992] Q. Dan, K.J. Dudek, "Certainty Factor Theory and its Implementation in a Medical Expert System Shell". *Medical Informatics*, vol. 17 (2), p. 87-103, 1992.

[Diaz and Haudenschild, 1983] O. Diaz and C. Haudenschild, "Implementation of an Integrated Critical Care Computer". *Computers in Critical Care and Pulmonary Medicine*, vol. 3, p. 297-312, 1983.

[Edelstien, 1990] J. Edelstien, "A study of Nursing Documentation". *Nursing Management*, vol. 21 (1), p. 42-53, 1990.

[Esfahani and Richardson, 1993] F. Esfahani and M. Richardson, "Impulsive Noise Removal From Measurement Data". *IMTC- Conference Record*, p. 608-611, 1993.

[Francois et al., 1992 (1)] P. Francois, B. Cremillieux, C. Robert, and J. Demongeot, "MENINGE: A Medical Consulting System for Child Meningitis. Study on a series of consecutive cases". *Artificial Intelligence in Medicine*, vol. 4 (19), p. 13-20, 1992.

[Francois et al., 1993 (2)] P. Francois, C. Robert, J. Astruc, J.L. Borderon, and D. Flovet, "Comparative Study of the Human Expertise and an Expert System: Application to the Diagnosis of Child Meningitis". *Computers and Biomedical Research*. Vol. 26, p. 383-342, 1993.

[Fumai et al., 1990] N. Fumai, C. Collet, M. Petroni, A. Malowany, F. Carnavale, and R. Gottesman, "A Simulator for an Intensive Care Unit Patient Data Management System". *Proceedings of the Third Annual IEEE Symposium on Computer Based Medical Systems*, p 454-461, 1990.

[Gallagher, 1988] N.C. Gallagher, "Median Filters: A Tutorial". *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 2, p. 1737-1744, 1988.

[Gardner, 1990] M. Gardner, "Patient-Monitoring Systems". In: *Medical Informatics*, chapter 12, p. 366-399, Addison-Wesley, Massachusetts, 1990.

[Gashing et al., 1983] J. Gashing, P. Klar, H. Pople, E. Shortliffe, and A. Terry, "Evaluation of Expert Systems: Issues and Case Studies". In: *Building Expert Systems*, Chapter 8, p.241-280. Addison -Wesley, Massachusetts, 1983.

[Georgeakis et al., 1991] D.L. Georgeakis, M. Evens, F. Naeymi-Rad, and D. Trace, "Performance Evaluation of Medical Expert Systems". *Computing in the 90s. The First Great Lakes Computer Science Conference Proceedings*, p. 70-76, 1991.

[Ginsberg, 1988] A. Ginsberg, "Knowledge Base Reduction: A New Approach to Checking Knowledge Bases for Inconsistency and redundancy". *Proceedings of the Seventh AAAI Conference*, p.585-589, 1988.

[Goss et al., 1995] E.P. Goss, V. Buelow, and H. Ramchandari, "The Real Impact of a Point-of-Care Computer System on ICU Nurses". *Journal of System Management*, vol. 46 (1), p.45-47, 1995.

[Green, 1982] J.F. Green, *Fundamental Cardiovascular and Pulmonary Physiology, An Integrated Approach for Medicine*. Lea & Fabiger, 1982.

[Gupta and Prasad, 1988] A. Gupta and B.E. Prasad, *Principles of Expert Systems*, IEEE Press, Washington DC, 1988.

[Hajek and Harmencova, 1995] P. Hajek and D Harmancova, "Medical Fuzzy Expert Systems and Reasoning about Beliefs". *Artificial Intelligence in Medicine, fifth Conference of AIME*, p. 403-404, 1995.

[Hammond et al., 1991 (1)] J. Hammond, H.M. Johnson, R. Varas, and C.G. Ward, "A Qualitative Comparison of Paper Flowsheets vs. a Computer Based Clinical Information System". *Chest*, vol. 99 (1), p.155-157, 1991.

[Hammond et al., 1991 (2)] J. Hammond, H.M. Johnson, R. Varas, and C.G. Ward, "Clinical Evaluation of a Compute 'Based Clinical Information System". *Heart and Lung: The Journal of Criti* ol. 20 (2), p.119-124, 1991.

[Hudson et al., 1984] D.L. Hudson, M.E. Cohen, P.C. Dadwania, and P.E. Watson, "Prospective Analysis of EMERGE, an Expert System for Chest Pain Analysis". *Proceedings of the 1984 IEEE Computers in Cardiology*, p19-24, 1984.

[Johnson and Aylor, 1988] B.W. Johnson and J.H. Aylor, "Reliability and Safety Analysis in Medical Applications of Computer Technology". *Proceedings of the IEEE Symposium on Computer Based Medical Systems*, p.96-100, 1988.

[Johnson, 1990] J.E. Johnson, "Nurse and Physician Perception". In: *Bringing Computers to the Hospital Bedside. An Emerging Technology*, Springer Publishing Company, New York, 1990.

[Jones, 1991] B.T. Jones, "Building Nursing Expert Systems Using Automated Rule Induction". *Computers in Nursing*, vol. 9 (2), p. 52-60, 1991.

[Karp and Wilkens, 1988] P.D. Karp and D.C. Wilkens, "An Analysis of the Distinction Between Deep and Shallow Expert Systems". *International Journal of Expert Systems*, vol. 2, p. 1-32, 1988.

[Keith et al., 1992] R.D. Keith, S. Beckly, J.M. Garibaldi, J. Westgate, and K.R. Grain, "A Multicenter Comparative Study of 17 Experts and an Intelligent System for Managing Labor". *Proceedings of the International Conference on Neural Networks and Expert Systems in Medicine and Health care*, p. 367-374, 1992.

[Keng and Bahill, 1990] Y. Keng and T. Bahill, "A Tool for Detecting Expert System Errors". *AI Expert*, p. 42-51, 1990.

[King and Smith, 1990] P.H. King and B.E. Smith, "An IBM AT Based Monitoring System With Touch Screen Input". *International Journal of Clinical Monitoring and Computing*, vol. 7 (2), p.107-111, 1990.

[Knight, 1990] J.C. Knight, "Issues of Software Reliability in Medical Systems". *Proceedings of the Third Annual IEEE Symposium on Computer Based Medical Systems*, p.153-161, 1990.

[Kolarz and Adlassing, 1986] G. Kolarz, K.P. Adlassing, "Problems in Establishing the Medical Expert System CADIAG-1 and CADIAG-2 in rheumatology". *Journal of Medical Systems*, vol. 10 (4), p. 395-405, 1986.

[Koski et al., 1990] E. Koski, A. Makvitra, T. Sukuvaraa, and A. Kari, "Frequency and Reliability of Alarms in the Monitoring of Cardiac Postoperative Patients". *International Journal of Clinical Monitoring and Computing*, vol. 7, p. 129-133, 1990.

[Kriewell and Long, 1991] T.J. Kriewell and T.M. Long, "Computer Based Medical Systems". *IEEE Computer*, vol. 24 (3), p. 9-12, 1991.

[Lam, 1993] A.K. Lam, "Development and Evaluation of an Expert System for an ICU". M. Eng. Thesis, McGill University, Canada, January 1993.

[Larson, 1992] J.A. Larson, *Interactive Software: Tools for Building Interactive User Interfaces*, Yourdon Press, 1992.

[Liebowitz, 1986] J. Liebowitz, "Useful Approaches for Evaluating Expert Systems". *Expert Systems*, vol. 5 (2), p. 86-96, 1986.

[Liu and Dillon, 1991] N.K. Liu and T. Dillon, "An Approach Toward the Verification of Expert Systems Using Numerical Petri Nets". *International Journal of Intelligent Systems*, vol. 6, p. 255-276, 1991.

[Loner, 1990] W. Loner, "Trends in User Interface Technology". *Proceedings of the ISA '90: Advances in Instrumentation and Control*, vol. 45, p. 1289-1291, 1990.

[Lucas and Jenssens, 1991] P.J. Lucas and A.R. Jenssens, "Development and Validation of HEPAR, an Expert System for the Diagnosis of Disorders of the Liver and Biliary Tract". *Medical Informatics*, vol. 16 (3), p. 259-270, 1990.

[Lundsgaarde, 1987] H.P. Lundsgaarde, "Evaluating Medical Expert Systems". *Social Science and Medicine*, vol. 24 (10), p. 805-819, 1987.

[Makvitra et al., 1991] A. Makvitra, E. Koski, T. Sukuvaraa, and A. Kari, "The Median Filter as a Preprocessor for a Patient Monitor Limit Alarm System in Intensive Care". *Computer Methods and Programs in Biomedicine*, vol. 34, p. 139-144, 1991.

[Meyer, 1992] C. Meyer, "Bedside Computer Charting: Inching Toward Tomorrow". *American Journal of Nursing*, p. 38-44, 1992.

[Milholland, 1988] K. Milholland, "Patient Data Management Systems: Computer Technology for Critical Care Nurses". *Computers in Nursing*, vol. 6 (6), p. 237-241, 1988.

[Miller and Sittig, 1990] P. Miller and D. Sittig, "The Evaluation of Clinical Decision Support Systems: What Is Necessary versus What Is Interesting". *Medical Informatics*, vol. 15 (3), p. 185-190, 1990.

[Miller et al, 1982] R.A. Miller, H.E. Pople, and J.D. Meyers, "INTERNIST-1 An Experiment Computer Based Diagnostic Consultant for General Internal Medicine". *The New England Journal of Medicine*, vol 307(8), p. 468-476, 1982 .

[Moritz, 1990] P. Moritz, "Information Technology: A priority for Nursing Research". *Computers in Nursing*, vol. 8 (3), p.110-115, 1990.

[Moser et al., 1990] L. Moser, W. Schramm, and G. Poser, "PC Based Computer Monitoring in an ICU". *Anesthetist*, vol. 39 (10), p. 561-564, 1990.

[Nenov et al., 1994] V.I. Nenov, E. Read, and D. Mock, "Computer Applications in the ICU". *Neurosurgery Clinics of North America*, vol. 5 (4), p. 811-827, 1994.

[Neuron Data, 1990] Nexpert Object User Guide, Neuron Data Inc., California, USA, 1990.

[Nguyen et al., 1987] T. Nguyen, W. Perkins, T. Laffy, and D. Pecora, "Knowledge Base Verification". *AI Magazine*, vol. 8, p. 69-75, 1987.

[Nguyen, 1985] T. Nguyen, "Checking an Expert Systems Knowledge Base for Consistency and Completeness". *Proceedings of the International Joint Conference on Artificial Intell gence*, p. 374-378, 1985.

[Nykennen et al., 1991] P. Nykennen, S. Choedhurry, and O. Wigertz, "Evaluation of Decision Support Systems in Medicine". *Computer Methods and Programs in Biomedicine*, vol. 34, p. 229-238, 1991.

[Paganelli, 1989] B.E. Paganelli, "Criteria for the Selection of a Bedside Information System for Acute Care Units". *Computers in Nursing*, vol. 7 (6), p. 241-221, 1989.

[Perdu and Luis, 1991] D.M. Perdu and A.H. Luis, "A Petri Net Model For Evaluating Expert Systems in Organizations". *Automatica*, vol. 27, p. 225-237, 1991.

[Peterson and Fisher, 1993] G.M. Peterson and P.D. Fisher," An Expert System for Pressure Ulcers: A work in Progress". *Canadian Conference on Electrical and Computer Engineering*, vol. 2, p. 973-976, 1993.

[Petroni et al., 1991] M. Petroni, N. Fumai, C. Collet, K. Roger, C. Yien, A. Malowany, F. Carnavale, and R. Gottesman, "An Automatic Speech Recognition System for Bedside Data Entry in an ICU". *Proceedings of the Fourth Annual IEEE Symposium on Computer Based Medical Systems*, p 358-365, 1991.

[Rennels and Miller, 1988] G.D. Rennels and P.L. Miller, "Artificial Intelligence Research in Anesthesia and Intensive Care". *Journal of Clinical Monitoring*, vol. 4, p. 274-289, 1988.

[Roger, 1992] K. Roger, "A Nursing Workload Manager for a Patient Data Management System". M.Eng. thesis, McGill University, Montreal, Canada, February 1992.

[Rossi-Mori et al., 1990] A. Rossi-Mori, D. Pisanelli, and F. Ricci, "Evaluation Stages and Design Steps for Knowledge-Based Systems in Medicine". *Medical Informatics*, vol. 15 (#), p. 191-204, 1990.

[Rousset, 1988] M.C. Rousset, "On the Consistency of Knowledge Bases: the COVADIS System". *Proceedings of the 8th European Conference on AI*, p. 79-84, 1988.

[Saab et al., 1995] E. Saab, V. Zia, O. Abu-Shihab, A. Malowany, F. Carnavale, and R. Gottesman, "A Data Model for a Patient Data Management System in an Intensive Care Unit". *Proceedings of the International Conference of Engineering in Medicine and Biology Society*, 1995.

[Saab, 1995] E. Saab, "A Database for an Intensive Care Unit". M.Eng. thesis, McGill University, Montreal, Canada, November 1995.

[Saarinen et al., 1991] K. Saarinen, P. Nykennen, K. Irjala, and M. Valimoki, "Design and Development of the Thyroid System interpreter". *Computer Methods and Programs in Biomedicine*, vol. 34, p.211-218, 1991.

[Safran et al. , 1990] C. Safran, F. Hermann, D. Rind, H. Kowaloff, and W. Slack, "Computer Based Support for Clinical Decision Making". *M.D. Computing*, vol. 7 (5), p. 319-322, 1990.

[Saranumi et al., 1991] N. Saranumi, T. Groth, A. Rosenfalk, and O. Wigertz, "Knowledge-Based Systems in Medicine - a Nordic Research and Development Program". *Computer Methods and Programs in Biomedicine*, vol. 34, p. 81-89, 1991.

[Scarpa et al.., 1992] R. Scarpa, S.C. Smeltzer, and B. Jasion, "Attitudes of Nurses Towards Computerization: A Replication". *Computers in Nursing*, vol. 10 (2), p. 72-79, 1992.

[Shiffman et al., 1991] S. Shiffman, A.D. Poon, C.D. Lane, and R.A. Miller, "Building a Speech Interface to a Medical Diagnostic System". *IEEE Expert*, p. 41-50, 1991.

[Shortliffe et al., 1981] E.H. Shortliffe, A.C. Scott, M.B Bischoff, and A.B. Campbell, "ONCOCIN: an Expert System for Oncology Protocol Management". *Proceedings of the Seventh International Conference on Artificial Intelligence*, vol 2, p. 876-881, 1981.

[Shortliffe, 1976] E.H. Shortliffe, *Computer Based Medical Consultation: MYCIN*, Elsevier/North-Holland, New York, 1976.

[Smith, 1992] V.J. Smith, "Monitor-Based Software Applications at the Bedside". *Computers in Nursing*, vol. 10 (2), p. 51-59, 1992.

[Solingen and Shabot, 1988] S. Solingen and M. Shabot, "A 32 Keyboard for the HP PDMS". *International Journal of Clinical Monitoring and Computing*, vol. 5 (1), p. 23-25, 1988.

[Staggers, 1988] N. Staggers, "Using Computers in Nursing: Documented Benefits and Needed Studies". *Computers in Nursing*, vol. 6 (4), p. 164-169, 1988.

[Stonham, 1991] G. Stonham, "Computing in Practice: Unfamiliar Language". *Nursing Times*, vol. 87 (29), p47-49, 1991.

[Stothert and Macleod, 1994] A. Stothert and I.M. Macleod, "Feature Extraction from Real-Time Expert Systems". *Engineering Applications of Artificial Intelligence*, vol. 7 (11), p. 361-366, 1994.

[Stotts and Kleiner, 1995] L. Stotts and H. Kleiner, "New Developments in Fuzzy Logic Computers". *Industrial Management and Data Systems*, vol. 95 (5) p. 13-17, 1995.

[Suen et al., 1990] C. Suen, P. Grogono , R. Shinghal, and F. Coallien, "Verifying, Validating and Measuring the Performance of Expert Systems". *Expert Systems With Applications*, vol. 1 (2), p. 93-102, 1990.

[Sun, 1993] K.W. Sun, "A Nursing Workload Scheduler for a Patient Data Management System". M.Eng. thesis, McGill University, Montreal, Canada, March 1993.

[Suwa et al., 1982] M. Suwa, A.C. Scott, and E.H. Shortliffe, "An Approach to Verifying Completeness and Consistency in Rule Based Expert Systems". *AI Magazine*, vol. 3, p. 16-21, 1982.

[Szolovits, 1995] P. Szolovits, "Uncertainty and Decisions in Medical Informatics". *Methods of Information in Medicine*, vol. 34, p. 111-121, 1995.

[Tachakra et al., 1990] S. Tachakra, D. Potts, and A. Idowu, "Evaluation of a Computerized System for Medical Records in an Accident Emergency Department". *International Journal of Clinical Monitoring and Computing*, vol. 7 (3), p. 187-191, 1990.

[Thompson, 1995] D.I. Thompson, "Reducing ICU Cost With Computerized Tools". *Health care Financial Management*, vol. 49 (6), p.66-70, 1995.

[Todd et al., 1993] B.S. Todd, R. Stamper, P. Macpherson, "A Probabilistic Rule-Based Expert System". *International Journal of Biomedical Computing*, vol. 33, p. 129-148, 1993.

[Turing, 1963] A.M. Turing, *Computers and Thought*. McGraw Hill, New York, 1963.

[Wick, 1992] M.R. Wick, "Expert Systems in Retrospect: A Case Study in the Evolution of Expert System explanation". *Journal of System and Software*, vol. 19, p. 159-169, 1992.

[Wyatt and Spiegelhalter, 1990] J. Wyatt and D. Spiegelhalter, "Evaluating Medical Expert Systems: What to Test and How ?". *Medical Informatics*, vol. 15, p205-217, 1990.

[Xue et al., 1993] W. Xue, S. Tielin, Y. Shugi, "Knowledge Base Verification". *Journal of Huazhong University of Science and Technology*, vol. 21 (1), p53-56, 1993.

[Yu et al., 1974] V.L. Yu, L.M. Fagan, S.W. Bennet, W.J. Clancy, J.F. Hannigen, and A.c. Scot, "Antimicrobial Selection by a Computer: A Blinded Evaluation by Infectious Disease Expert". *Journal of the American Medical Association*, vol. 24 (2), p. 1279-1282, 1974.

[Zhang and Nguyen, 1994] D. Zhang and D. Nguyen, "PREPARE: a Tool for Knowledge Base Verification". *IEEE Transactions on Knowledge and Data Engineering*, vol. 6 (6), p. 983-989, 1994.

[Zia, 1996] V. Zia, "Nursing Workload Management in Intensive Care". M.Eng. thesis, McGill University, Montreal, Canada, *to be submitted* June 1996.