A Model-Based Expert System for Interpretation of Hemodynamic Data from ICU Patients

by

Ruilin Zhao

May 1997

This research was supported by an An Wang Graduate Fellowship, a Medical Engineering and Medical Physics (MEMP) Graduate Fellowship from Harvard-M.I.T. Division of Health Sciences and Technology, and a research grant from Hewlett-Packard Company.

A Model-Based Expert System for Interpretation of Hemodynamic Data from ICU Patients

by

Ruilin Zhao

This report is a modified version of a thesis submitted to the Department of Electrical Engineering and Computer Science on May 18, 1997, in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering and Computer Science

Abstract

With the proliferation of modern monitoring and laboratory procedures, physicians in intensive care areas may face "information overload", in dealing with very large, complex and ever-changing quantities of clinical data, which often lacks efficient organization. This research analyzes the medical knowledge required for formulating decision models in the domain of hemodynamics. Based on such analysis, a knowledge based expert system to track a patient's hemodynamic state has been developed and evaluated in a laboratory setting.

The initial phase of the work utilizes a cardiovascular simulator to generate "pseudo-ICU" waveforms as input to the expert system in order to guide the development of the matrix of rules and search strategies. A number of pathological simulations have been successfully analyzed by this model-based expert system, including examples of hypertension, left ventricular failure, hypovolemia, pulmonary hypertension, etc. We conclude that our approach is practical, and provides a mechanism for transforming and reducing real-time physiologic data into pathophysiologic hypotheses relevant to the management of patients.

Thesis Supervisor: Roger G. Mark
Title: Professor of Electrical Engineering and Computer Science

Thesis Supervisor: Peter Szolovits
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I came to this country three years ago. During this period, I have been exposed to the American's culture and society. I had some difficult times as you can imagine. Fortunately, I have so many people who cared about me. I have many to thank for guiding me towards this thesis.

My advisor, Professor Roger G. Mark, has always been very helpful at the most difficult time in this period. He has always been very patient with me. He is always available to talk to me. He is the person to whom I always like to talk if I have any kind of problem.

My co-advisor, Professor Peter Szolovits, has long supplied thoughtful, valuable advice to my research as well as to my life. His famous quote from German philosopher Friedrich Nietzsche, "What does not kill me makes me stronger," has been very inspiring to me spiritually.

George Moody, a research scientist in Dr. Mark's group, has been a constant source of advice on my research work.

My host family in America, Coit's family, has been very helpful all the time. They are the people whom I can run into and just ask some silly questions. They gave me more than I deserved. Their summer house in New Hampshire has always been a nice place to visit.

Members past and present at the Clinical Decision Making Group at M.I.T., have made my graduate student life continuously stimulating and enjoyable. My office mate, Christien L. Tsien, has helped me in many ways she could. The other people in the *medg* group, Latanya Sweeney, Eric Jordan, Hamish Fraser, Sun Yao, and Milos Hauskrecht are good colleagues to work with. Very special thanks to Michael Frank for his kindness in debugging the language problems in this thesis.

My family, although physically is far away from me all the time, has given me unbounded love and support throughout my academic career. They share with me the sense of accomplishment this thesis has brought. Most especially, my father and my mother have helped me spiritually. My sister and my brother have offered me their love, support and care. My two nephews are growing up; that is also a inspiring thing to know and think about all the time.

This research has been supported by an An Wang Graduate Fellowship, a Medical Engineering and Medical Physics (MEMP) Graduate Fellowship from Harvard-M.I.T. Division of Health Sciences and Technology (HST), and a research grant from Hewlett-Packard Company. Special thanks to the Wang Company, the HST Division and the Hewlett-Packard Company.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Intensive care units (ICU) provide continuous and often invasive measurements of respiratory and hemodynamic status in acutely ill patients. Such monitoring enables early detection of changes in the patient's condition and provides information that both directs therapy and assists in evaluating the response to treatment. Needless to say, it generates enormous amounts of real-time and off-line data relating to the status of these patients. The ICU medical staff must re-assess these patients frequently on the basis of data from clinical observations, bedside monitors, mechanical ventilators and a wide variety of lab tests. Clinicians usually face a very large, complex and ever-changing body of data, and it is often challenging and time consuming for them to analyze such massive information loads. As we can imagine, with the development of new techniques and medical equipment, the problem of *information overloading* will become more and more serious. The great responsibilities of ICU staff and high performance expectations can lead to both emotional and physical fatigue [39]. All of these factors may cause errors in patient care. Providing life support in the ICU is becoming an increasingly complex task as the volume of monitoring data increases. Improvement in the organization and interpretation of the clinical data will have the potential of increasing quality and efficiency in the ICU.

Most intensive care units have the basic capacity to monitor and record heart rate and rhythm (usually with a built-in memory and recall capability), venous pressure, pulmonary arterial pressure, and systemic arterial pressure. In addition, many units have the instruments necessary for measurement of cardiac output. In this thesis, I design and implement a knowledge based system which interprets simulated ICU hemodynamic data. First, multiple ICU measurable data are gathered from a cardiovascular simulator, including central venous pressure, arterial blood pressure, pulmonary artery pressure, ventricular pressure, cardiac output, etc. Then I determine the meaning of these measurements with reference to each pseudo-patient's clinical problem, and suggest appropriate settings for the systemic parameters of the cardiovascular simulator to simulate the state of the pseudo-patient. The system is intended to help physicians to diagnose and track the disease of patients.

Thorough analysis of many clinical decision making systems has given us a clearer picture of symbolic approaches to medical decision making. Additional requirements for decision aids in intensive care areas include programs which can handle data that are changing over time in order to provide tracking of the patient's status during the course of the underlying disease or in response to therapeutic intervention. The long term objective of this research is to understand how we can manage ICU data in a more systematic way, and represent the patients status as a set of parameters in a cardiovascular simulator. We seek to develop methods to improve general model-based reasoning, and try to find a quantitative or at least a semi-quantitative reasoning method.

1.1 Artificial Intelligence

The origin of artificial intelligence can be traced back to the earliest days of machine computation [16]. In 1843, Ms. A. A. Byron raised the question of whether Babbage's proposed analytical engine, the first programmable computing machine, might "think". Although not called "artificial intelligence" at that time, lots of work was done on machine translation beginning in the early 1950s. It was in 1956 that artificial intelligence began as a separate aspect of computer science at the famous Dartmouth conference [5]. This period was named the *prehistory* period in [13], because few ideas could be experimentally tested. It extended to about 1960 because there was no adequate computers available then.

Around 1960, the *Dawn Age* [13] started. People tended to believe that in ten years, they could make computers as "smart" as people are. The Dawn Age was marked by some limited successes. Patrick Winston named the period from around 1965 until about 1970 the *Dark Period* [13], because there was almost no progress.

Then from about 1970 to 1975, people started to think about how to impress the public, industry, etc. That was when some application systems, such as MYCIN, came out. It was called *Renaissance* [13]. This was followed by the *Age of Partnerships* [13], a period when researchers started to pay attention to the research fruits coming out from other domains and to form some important liaisons.

Then from 1980 till about 1990, it was *Age of the Entrepreneur* [13]. During that period, lots of start-up companies focused on designing possible intelligent systems. After 1990, with the failure of almost all of those intelligent-wanna-be companies, I would like to call our present age the *Age of Adjustment*. The name comes from my analogy to the stock market. We cannot expect that a tree can grow to the sky. The potential and the expectation have been oversold by the media and the public. Now it is time to come back to re-adjust our positions and expectations for both long term and short term.

There has been tremendous progress in some of the subfields of artificial intelligence during the past several decades. Some successes are very domain specific, such as: XCON (or R1) developed by Digital Equipment Corporation and Carnegie Mellon University for doing computer configuration; PUFF developed by Stanford University for analyzing certain lung problems. Other successes are less domain specific, such as INTELLECT, a powerful interface between decision makers and databases.

But the problem is just what Gerald Sussman et al. said [3], "The field overall ... is moving increasing attention to near-term applications, retarding progress toward comprehensive theories and deep scientific understanding, and ultimately, retarding progress toward developing the science needed for higher-impact applications." Part of the reason for people to focus on near-term applications is the growing anxiety and lack of trust in AI research among the public. As a result, researchers are very eager to show some progress to the public to make them feel more comfortable and confident about artificial intelligence. If I may make another similar analogy, we can say that the share holders (*public*) are too desperate about artificial intelligence. But we can not blame the general public too much, because I think part of the reason for people to have such high expectations is that some researchers have made and continue to make rash predictions of rapid progress in the near future.

An interesting criticism comes from some neuroscientists who claim that computer scientists want to make a machine, but they don't even know how the machine works. In their opinion, most artificial intelligence researchers should either find new jobs or go home to sleep and wait until they figure out the mechanisms of the brain. An easy counter-argument is that we did not have to understand aerodynamics to make a plane.

Other criticisms come from some philosophers who claim that one cannot make another equally intelligent system with oneself, just like one cannot lift oneself, no matter how strong s/he is.

Personally, I am optimistic about artificial intelligence in the long term, but pessimistic in the short term. Some people even express the concern that artificial intelligence is *the biggest joke in the 21st century*. The history of the development of science tells us that AI is very unlikely to be a joke. Now it is time for us to focus on solving some fundamental questions of artificial intelligence. Shimon Ullman et al. said [3], "Traditional approaches to artificial intelligence have been largely organized around unconstrained algorithms or simplistic biological models. Both lack of constraints and incorrect constraints have imposed show-stopping penalties." Many biologists believe that the *neural network* in computer scientists' dictionary should be erased or changed, because that is an over-simplified model for neurons. Then should we all go back to the bench and look for a more complicated biological model? It is easy to measure the detailed activity of a single neuron, but measuring the systemic activity of a neural region is very difficult because we lack the needed powerful computer hardware and bio-techniques. One thing I can be sure of is that we need concrete, not superficial, cooperation between scientists in different fields. We need some team work. It is very naive to try to form some national team to cope with this problem, because after all this is not the same case as fighting some epidemic disease like AIDS. But forming some regional or institutional cooperative teams is certainly feasible. Gerald Sussman, Patrick Winston, Shimon Ullman Kenneth Yip, and other colleagues have recently formed a team to try to *redirect* our research toward a program focused on understanding natural intelligence and building integrated intelligent systems. To conclude my thoughts on prospects for artificial intelligence, I will say: "The light is on the way, but far away."

1.2 Knowledge-Based System

There are three major research issues concerning the knowledge based paradigm [6]:

- Knowledge Representation: In which way should knowledge be represented as symbolic data structures for computer use?
- Knowledge Utilization: What designs are available for the inference procedure?
- Knowledge Acquisition: How can we develop a systematic way for computers to access the knowledge?

The earliest knowledge based systems date back to the mid 1960's [35]. One of the conspicuous systems was MACSYMA, developed by the Mathlab Group at Project MAC at M.I.T. MACSYMA can be used to provide a very broad range of symbolic computing capabilities, such as integration, power series and support for manipulating matrices [35]. Another major early expert system is DENDRAL, developed by Stanford University and used to determine the three dimensional structure of a chemical compound from its chemical formula and some other information.

The success of those early expert systems boosted the development of other expert systems. In [10], Feigenbaum reported that there were about 1500 expert system applications in actual use and a few thousand more in field testing. Despite the broad range of applications and different domains, these systems focus on several central tasks [35]:

- Diagnosis: Focus on how to get a reasonable explanation for some abnormal values of observables. Examples include medical diagnosis, etc.

- Selection: Focus on how to make intelligent selection. Examples include insurance, credit authorization, etc.

- Configuration and planning: Focus on how to complete a partial design or adjust a complete design to meet some constraints. Examples include XCON.

Parallel to the development of artificial intelligence, knowledge-based systems are also facing some fundamental problems [35]:

- Fragility of Encoded Knowledge: How to incorporate common sense knowledge into the system? How to avoid common-sense-violating behavior? One of the biggest issues here is that human beings have too much common sense to explicitly encode in computers.

- Complexity in the Real World: Many systems work well for straightforward problems but fail on the more complicated ones.

- Sources of Better Models: In many domains, we lack the ability to capture or design the depth and breadth of the domain knowledge needed to allow the computer to produce expert behavior.

Reviewing these problems, we may feel that we actually know *too little* about ourselves, about our learning process, about our knowledge storage forms, etc. Do we believe that the storage form of the knowledge in our brain is kind of synapse connection, chemical compound diffusion or something else? One may argue that probably the reason why the computer is having problems with common sense and complexity is that the computer was made too fast to use. Human beings develop common sense after living in society for about 20 years. Imagine a scenario, in which we put a computer with a new-born kid together, and each time when we teach this kid something, we will try to send the same knowledge to the computer. Would this be enough to develop an intelligent machine? But the major problem here is that most of the time, we will have trouble *teaching* computer to learn something new and abstract. Typically for the common sense knowledge, we do not know how to extract the features of the common sense knowledge.

1.3 Artificial Intelligence in Medicine

It has long been recognized that computers could be put to beneficial use in the medical field. As early as the mid 1950's, physicians and computer scientists had recognized that computers could be used to assist in making clinical decisions and potentially to implement automatic decision making systems. There are several general approaches [13]:

- Flowcharts: Encode a serial description of actions which good physicians will take.
- Large Clinical Data Base: Search among previously studied cases and match the current case.
- Probability Theory: Based on the observable data, derive the probabilities of several explanations.

But starting in the 1970's, scientists involved in computer-based reasoning began to recognize the potential benefits of applying symbolic reasoning techniques in clinical domains [33]. The first medical reasoning program, known as the MYCIN System [32], adopted symbolic processing techniques largely in response to a conviction that computer-based consultation systems, in order to be accepted by physicians, should be able to explain how and why a particular conclusion had been derived [32]. Subsequently, a series of additional medical application programs have been created. For example, the PIP [28] (Present Illness Program) is designed to simulate human clinical cognition and the INTERNIST [27] diagnostic system of Pople and Myers is a computerized diagnostic program which emphasizes a very broad coverage of clinical diagnostic situations. These systems were impressive, but their use revealed many more problems, such as inadequate criteria for deciding when a diagnosis is complete [36], and the need for better representation of anatomical and physiological medical knowledge [17].

Although several expert systems were developed for medical problem solving during the past couple of decades, AI has failed to make a significant impact on health care delivery [37]. First of all, most initial efforts in artificial intelligence in medicine (AIM) were focused on developing automated diagnosis systems, which make medical professionals nervous and unwilling to cooperate [37]. Other reasons are the complexity of medicine itself, including the complexity and inherent variability of human anatomy and physiology, and our lack of understanding of the medical decision process on a cognitive level. With the general dim outlook of artificial intelligence, AIM is also facing its greatest challenge. We must make some applicable systems to please society. Now AIM is focusing to a greater extent on knowledge management and intelligent advice-giving instead of the ambitious task of automated diagnosis. It is much easier to generate some advice than to replace physicians, both scientifically and socially.

In the past several years, there has been a rise in AIM activity. Some of the reasons for the increased activity are the increase in computer power and a rise in physicians' interests in computer technology. Due to the increasing demands of information processing that continue to challenge physicians, they have come to demand computer technology for patient care purposes. Therefore, the development of methods to manage endless streams of data and to assist in making decisions using these vast sources of information are among the top priorities of AIM researchers [32]. On the other hand, significant progress has been made in model-based reasoning, mainly due to the development of theories and methods for qualitative modeling of physical devices and systems. Several model-based computer programs have been devised in medical AI, such as AI/MM [20] (dealing with a mathematical model of renal physiology), LOCALIZE [11] (dealing with the localization of lesions within the peripheral nervous system), The Heart Failure Program [21] (reasoning about the function of the cardiovascular system), etc.

The current problem of AIM is that with our continued lack of fundamental understanding of the reasoning process, how far can AIM go? If we know that intelligence can be represented or replaced by pure calculation, then we believe, theoretically, that computers can replace or simulate the thinking process of human beings. Even so, applying artificial intelligence techniques to the medical system would bring some extra concerns which do not exist in other domains:

- Cost of the Computation: Obviously, we must set a time limit for running computers when we need real-time responses. For example, we can not wait a day for the computer's results in ICU areas.

- Legitimization: How to let those intelligent pseudo-doctors go through the legitimization process? If the wrong diagnosis happens, who will be responsible?

- Social Acceptance: Patients may just refuse to be seen by those intelligent pseudo-doctors, because they feel more comfortable in dealing with physicians or nurses personally.

1.4 Background and Motivations

Twenty years ago, ICU care focused solely on the assessment of vital signs and laboratory values. Today, routine intensive care involves a great variety of monitoring devices, infusion pumps, drug-administration systems, sophisticated ventilators, etc. The responsibilities of ICU staffs are complex [31]; they usually assess more than 50 measurements, laboratory values, and physical findings, etc. Also, most of the current medical monitoring systems include built-in alarms, and permit users to change the alarm limits. However, these automated alarms operate independently, and are so unaware of the clinical context that they have a high rate of false alarms. ICU staffs may delay their response to alarms that are frequently in error, or even ignore them [24].

The large amount of monitored ICU data may create information overload, which leads to errors and mishaps in ICU care. There are occasional tragedies, most of which are due to human error, reported in ICU care. In [1], Abramson estimated that about 90 of 150 adverse occurrences in a surgical ICU were caused by human error. We may also find the similar results regarding human errors which happened in ICU care [29]. So we may expect that with the development of medical instruments, more and more clinical data are going to be presented to ICU staffs. So information overload will become an increasingly serious problem unless better methods are designed to organize and present the data to the clinicians. Relative to the rapid development of medical instruments and tests, the slow progress of information process and synthesis may impede the overall advancement of health care. Fortunately, computer based access to ICU data has made it possible for researchers to analyze large databases and to create statistical prediction models which can be used to qualitatively evaluate and predict the status of patients, such as APACHE (Acute Physiology, Age, Chronic Health Evaluation), which predicts short-term survival in the ICU.

So far, most of the intelligent alarm systems have failed to interpret the meaning of changes of multiple physiologic variables for the overall clinical context. Those alarms usually only highlight important events. To perform an overall clinical context-based interpretation will require a model for ICU patients, so that we can compare the current status of patients with respect to their previous status in an overall clinical context, instead of only focusing on one or two physiological signals.

1.5 Related Work

In the early 1980's, Larry Fagan developed an expert system called VM (Ventilator Manager). VM is designed to interpret on-line quantitative data in the ICU [9]. It was able to produce interpretation of physiological measurements over time. Since then, a number of approaches have been developed to interpret ICU data [12]. In intensive care areas, automated methods that monitor the hemodynamic status of patients must reason about and respond to changes in large amounts of clinical data, such as blood pressure, heart rate, cardiac output, etc. Many researchers have developed techniques to interpret ICU data [12]. Generally, we can divide these techniques into two categories: numeric and symbolic methods. The numeric methods are reasoning processes for providing quantitative analysis; and symbolic methods deal with qualitative analysis. In intensive care areas, the clinical context includes data that are both numeric (such as that the arterial blood pressure is 120/80 mmHg) and symbolic (such as the interpretation of an echocardiogram). Other programs included qualitative information in the interpretation of patient data by applying rule-based symbolic models. The GUARDIAN project, for example, implements an adaptive intelligent agent to reason about monitored ICU data within the real-time constraints of ICU care. Mathematical models are also powerful tools for simulating the quantitative time-dependent behavior of complex, dynamic physiologic systems. Mathematical models that include physiologic concepts and feedback mechanisms allow researchers to study interactions among physiologic systems.

We believe that qualitative and semi-quantitative analyses for patients in the ICU are very important when we try to estimate the parameters that describe a patient's physiologic state. We developed a method for finding patient-specific model parameters of detailed mathematical models. The patient-specific model parameters will be presented to the physicians to illuminate the fundamental pathology and to suggest patient-specific treatment recommendations.

1.6 Overview of Thesis

This introductory chapter has briefly discussed the development and the current status of artificial intelligence, knowledge-based systems and the development of the specific domain — artificial intelligence in medicine. We talked about the background and the motivations of this project, and also gave some background about related work.

The remainder of this thesis is organized as follows:

- Chapter 2 describes the medical domain associated with this work, that is ICU patient care. Specifically, it describes some cardiovascular patients. An example case is also included for illustration.

- Chapter 3 describes the tool, a cardiovascular simulator, used in the research. It gives an overall picture about the structure of the hemodynamic model and baroreflex model, which are used in the simulator. Some example cases to simulate cardiovascular patients are also included.

- Chapter 4 details the analysis of the reasoning model. First, we talk about the homotopy method, which is the method we used when we started the research. Then the method used in our current system is elaborated.

- Chapter 5 describes the evaluation of the system. What is our ideal system? How do we evaluate its performance? Finally, we lead to the discussion of the question — What is acceptable performance?

- Chapter 6 lists the accomplishments, limitations, practical considerations and future directions of this work.

Chapter 2

Domain Knowledge

An intensive care unit is defined by its ability to provide the enviroment, facilities, and personnel for the care of severely ill patients. The important features required of such a unit are listed as follows [39]:

1. High nurse-to-patient ratio
2. Ready accessibility of physicians
3. Ability to provide invasive hemodynamic monitoring
4. Availability of respiratory support techniques
5. Ability to provide supervised intravenous infusions of pharmacologic agents

In severely ill patients, the interrelationship between different physiological systems must be noticed. ICU staff should not focus on one component of the illness, even if that component is predominant and may yield a therapeutic approach. Usually, management of patients in the ICU presents lots of dilemmas. For example, treatment directed to reduce the pulmonary capillary wedge pressure in patients with adult respiratory distress syndrome may adversely affect renal and central nervous system perfusion. On the other hand, increasing the pulmonary capillary wedge pressure to optimize cardiac output in a patient with ischemic heart disease may result in non-cardiogenic pulmonary edema. In most cases, ICU staff must synthesize an overall diagnostic and treatment strategy that considers the views of various specialists with a narrower focus and then make some compromise.

2.1 ICU Patient Care

Most ICU areas have a very general orientation, treating almost all types of severely ill patients. Because of the sickness of ICU patients, ICU units require clear delineation of administrative and medical guidelines of authority and responsibility. Therefore, a general guideline for admission and discharge of patients is required. The existence of such policies reduces the apparent ambiguity often inherent in the difficult environment of a ICU and enables prompt decision making by ICU staff [39].

An ICU usually can provide continuous and often invasive measurements of respiratory and hemodynamic status for severely ill patients. The complexity of monitoring systems varies considerably, it can range from simple electrocardiographic monitoring with only a real-time screen display to automated "closed loop" systems to regulate intravenous infusions of fluids and drugs.

The complications of intensive care usually are difficult to detect from the illness [39]. For example, cardiac arrhythmias and gastrointestinal hemorrhage are very common in ICU patients, but they may or may not necessarily relate to the care in the ICU. However, some complications are straightforward and clearly related to an intervention taking place in an ICU. For example, ventricular tachycardia (VT) can occur during the passage of a Swan-Ganz catheter through the right ventricle.

In addition to patient-related complications, the ICU care environment can also stress ICU staff psychologically. There have been many reports that the negative psychologic effects of working under great high pressure conditions can cause ICU staff "burn-out".

2.2 Respiratory Monitoring

Usually, as a minimum, respiratory monitoring should include measurement of respiratory rate and periodic measurement of pressure of oxygen ($P_{a_{O_2}}$), pressure of carbon dioxide ($P_{a_{CO_2}}$), and arterial blood pH. Respiratory rate can be measured and recorded automatically in non-intubated patients using impedance devices. Both respiratory rate and tidal volume delivered by the ventilator (V_T) can be monitored in intubated, spontaneously breathing patients using a spirometer and appropriate alarms [39].

For the mechanically ventilated patients, it is essential to monitor respiratory rate, exhaled V_T, and airway pressure. We may also use other monitoring techniques, such as breath by breath measurements of respiratory system compliance, volume-pressure, volume-flow relationships, and measurements of $F_{I_{O_2}}$ and exhaled carbon dioxide and oxygen. These measurements may provide some useful information such as early indications of changes in $P_{a_{CO_2}}$, but their usefulness remains uncertain.

The transcutaneous P_{O_2} and P_{CO_2} that are indirect reflections of $P_{a_{O_2}}$ and $P_{a_{CO_2}}$ can be measured continuously using heat skin electrodes. The measurement of P_{O_2} remains uncertain for adults. Pulse oximeters are noninvasive monitoring devices that measure and record oxyhemoglobin saturation. This measurement is useful in some clinical situations, such as evaluating oxygenation during sleep and during bronchoscopy procedures. Indwelling catheter electrodes for measurement of $P_{a_{O_2}}$ and $P_{a_{CO_2}}$ and arterial pH have been used but have some technical limitations [39].

There are four major complications related to respiratory support in ICU areas [39]:

1. **Oxygen Therapy** is administered by external devices. It may cause the following side effects:

    - discomfort related to the device or to administration of dry gas, which can be managed by changing the device or improving the humidification
    - fires and hypoventilation because of uncontrolled administration of oxygen in patients

2. **Artificial Airways** are associated with a number of complications. The immediate and long term injury to nose, hypopharynx, larynx or trachea, can be caused by an endotracheal tube. It may also cause pulmonary infection.

3. **Tracheostomy** avoids the problems of an endotracheal tube because it bypasses the upper airway. However, tracheostomy has more disadvantages than advantages. Problems can include hemorrhage, mediastinal and tracheal stenosis. It may also cause pulmonary infection.

4. **Mechanical Ventilation** can have overventilation and underventilation problems. Oxygen toxicity usually occurs in patients being mechanically ventilated with gas mixtures containing high concentrations of oxygen [39].

2.3 Hemodynamic Monitoring

Since our project focuses on the interpretation of hemodynamic data, a detailed description of hemodynamic monitoring in the ICU is worthwhile. Most ICUs have the basic capacity to monitor and record heart rate and rhythm, venous pressure, pulmonary arterial pressure, and systemic arterial pressure. Many units also have the instruments to measure cardiac output.

Clinically, the following monitoring is usually done in ICU areas [39]:

1. **Electrocardiographic Monitoring** is of great value in patients with specific cardiac disorders such as acute myocardial infarction or cardiac arrhythmia. Also, because it is easy and relatively inexpensive, ECG monitoring has been widely used in ICU areas. The automated detection of heart rate and rhythm is generally an integral part of the monitoring systems. Abnormalities in heart rate or rhythm may signal some worsening of respiration, electrolyte abnormalities, and other noncardiac problems.

2. **Systemic Arterial Pressure Monitoring** is obviously of great value. Changes in blood pressure can be detected immediately and therefore can enable *beat by*

beat assessment of the effects associated with the changes in ventilatory pattern. Note that beat to beat variations in systemic arterial blood pressure may not warrant specific intervention. The usual technique to monitor systemic arterial blood pressure is to insert percutaneously a Teflon catheter into an accessible artery. The catheter is connected through a stopcock to a rigid connecting tube that in turn is attached to a transducer.

3. **Central Venous Pressure Monitoring** is useful in assessing the course of right ventricular failure, right ventricular infarction and tricuspid regurgitation. The usual technique to monitor central venous pressure is to insert percutaneously a catheter into either the subclavian or the external or internal jugular vein.

4. **Pulmonary Arterial Pressure** can be measured by a Swan-Ganz catheter. This catheter has a balloon just proximal to the tip and a separate lumen for inflating the balloon. The Swan-Ganz pulmonary catheter can provide measurement of pulmonary arterial blood pressure (P_{PA}), pulmonary arterial wedge pressure (P_{PAW}), central venous pressure (CVP) and cardiac output. However, there are some problems with the measurement of P_{PA}. Because of the oscillations in pleural pressure on the measured intravascular pressure, the P_{PA} signal must be carefully analyzed in order to obtain an accurate value.

5. **Cardiac Output** can be obtained using the thermistor equipped Swan-Ganz catheter. In addition, the instrumentation for performing thermal dilution process and a computer for processing dilution measurements and calculating cardiac output are required.

Based on the above measurements, ICU staff then use Table 2.1 to guide an understanding of a patient's situation.

There are three principal types of problems associated with hemodynamic monitoring: local complications related to vascular access, passage and final positioning of the catheter, and inappropriate decision making based on inaccurate data or in-

Situation	$\bar{P}_{SA}$	$\bar{P}_{RA}$	$\bar{P}_{PA}$	$\bar{P}_{PAW}$	$C.O.$	PVR	SVR
Hypovolumic Shock	↓	↓	↓	↓	↓	↑	↑
Septic Shock	↓	↓	↓	↓	↑	↓	↓
Cardiogenic Shock	↓	↑	↑	↑	↓	↑	↑
Pulmonary Embolism	↓	↑	↑	→↓	↓	↑	↑
Airway Obstruction	→	→↑	↑	→	→	↑	→
Right Ventricular Infarction	↓	↑	→	↓→↑	↓	→	→↑
Cardiac Tamponade	↓	↑	↑	↑	↓	→	↑
End Stage Liver Disease	↓	→↓	→↓	→↓	↑	→	↓

Table 2.1: Patterns of Hemodynamic Abnormalities in Severely Ill Patients, from [39].$\bar{P}_{SA}$: Mean Systemic Arterial Pressure; $\bar{P}_{RA}$: Mean Right Atrial Pressure; $\bar{P}_{PA}$: Mean Pulmonary Arterial Pressure; $\bar{P}_{PAW}$: Mean Arterial Wedge Pressure; $C.O.$: Cardiac Output; PVR: Pulmonary Vascular Resistance; SVR: Systemic Vascular Resistance

complete data or misinterpretation of information from the monitoring device. The usual complications related to hemodynamic monitoring are listed as follows:

- Systemic Arterial Pressure: The most frequent complication related to SAP monitoring is the formation of a hematoma at the site of the arterial puncture. The hematoma may result in the peripheral nerve damage.

- Central Venous Pressure: The complications of CVP monitoring include air embolism, infection and venous thrombosis.

- Pulmonary Arterial Pressure: All of the complications associated with CVP monitoring can happen here. In addition, PAP monitoring can cause a disturbance of cardiac rhythm or conduction or both. Premature ventricular contractions occur very commonly when the catheter passes through the right ventricle.

2.4 An Example Case

An example case in the domain of ICU patients with severe cardiovascular problems is shown below [7]:

The patient is a 37 year old woman with a positive history for HIV and intravenous (IV) drug abuse. She was brought into an ICU with a fever of 104F, extreme lethargy, *scant urine output, and a tender abdomen, especially in the right upper quadrant. She was noted to be tachycardic, hypotensive, and tachypneic. After being accepted to the ICU, her systemic arterial pressure, central venous pressure, right ventricular pressure, pulmonary arterial pressure, and pulmonary arterial wedge pressure were continuously recorded for assessment.*

In the following chapters, we will examine the process for automated diagnosis and propose an iterative method for setting the systemic parameters of a cardiovascular simulator to simulate the status of ICU patients.

Chapter 3

Cardiovascular Simulator

Automated methods that provide treatment advice require a model of physiology to assist them in the interpretation of ICU patient data. A lumped-parameter cardiovascular model is used in our expert system to represent the patient's hemodynamic state.

The cardiovascular simulator (CVsim) is a dynamic computer simulator of human cardiovascular hemodynamics [7], originally intended for students of physiology and medicine in Harvard-MIT Division of Health Sciences & Technology (HST). It is implemented on SUN workstations running the X Window System, allowing students to perform a variety of measurements in a real-time simulated cardiovascular system, not all of which would be possible in an animal laboratory. The simulation is based on the lumped-parameter mathematical model taught in the Cardiovascular Pathophysiology course (HST090) in HST.

The model is shown in Figure 3-1. It includes four major components: left heart, systemic circulation, right heart, and pulmonary circulation. Each side of the heart is modeled by a variable capacitor (representing the pumping action of both atrium and ventricle), two diodes representing the AV and arterial valves, and outflow resistances. There are 23 parameters for defining the status of the simulator; their normal values are shown in Table 3.1.

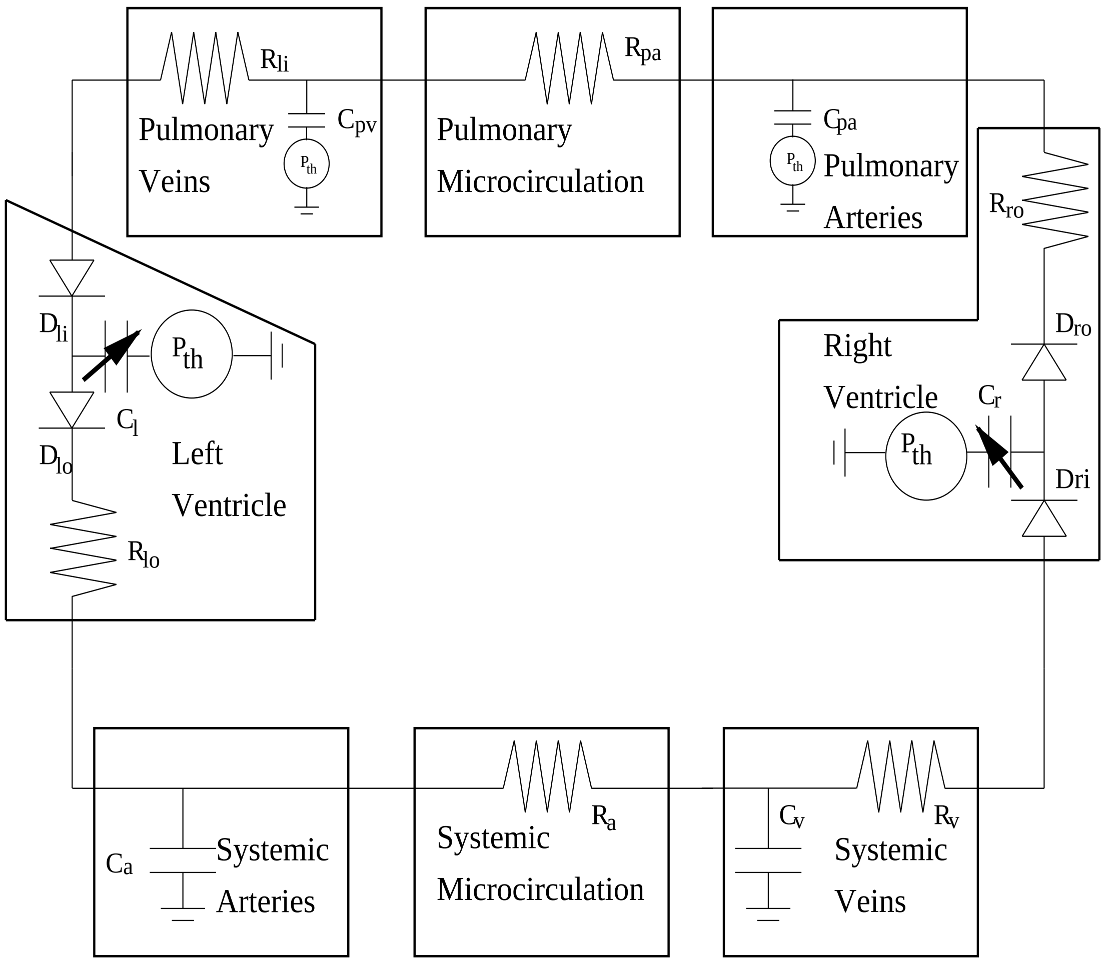

Figure 3-1: Circuit Diagram Equivalent of Lumped Parameter Model

3.1 Background

Several cardiovascular system models have been put forward since 1950's [23]. In 1959, the first model of the cardiovascular system was formulated by Grodins [14]. In that model, he defined the steady state operation of the ventricle, then combined two ventricles with a pulmonary circuit branch to simulate the effects of Starling heart-lung relationship. This work was considered as the starting point for integrative physiology [7]. Then in early 1960's, a six node electrical circuit model was proposed by Defares and his colleagues [8]. In that model, each node consists of a capacitor to ground and is connected to two adjacent nodes by impedances. The varied left and right ventricular capacitances were used to simulate the time varying elastance property of contraction. Early in 1970's, Dr. Arthur Guyton developed refined com-

Parameter	Normal Value
Heart Rate	72beats/min
Total Blood Volume	5000ml
Transthoracic Pressure	-4mmHg
Capacitances:(ml/mmHg)	
Left Ventricular Systolic Capacitance	0.4ml/mmHg
Left Ventricular Diastolic Capacitance	10.0ml/mmHg
Right Ventricular Systolic Capacitance	1.2ml/mmHg
Right Ventricular Diastolic Capacitance	10.0ml/mmHg
Systemic Arterial Capacitance	1.5ml/mmHg
Venous Capacitance	100.0ml/mmHg
Pulmonary Arterial Capacitance	4.3ml/mmHg
Pulmonary Venous Capacitance	8.4ml/mmHg
Zero-Pressure Volumes:(ml)	
Left Ventricular Zero-Pressure Volume	15.0ml
Right Ventricular Zero-Pressure Volume	15.0ml
Systemic Arterial Zero-Pressure Volume	715.0ml
Systemic Venous Zero-Pressure Volume	2500.0ml
Pulmonary Arterial Zero-Pressure Volume	90.0ml
Pulmonary Venous Zero-Pressure Volume	490.0ml
Resistances:(PRU=mmHg*sec/ml)	
Systemic Arterial Resistance	1.0PRU
Systemic Venous Resistance	0.01PRU
Pulmonary Resistance	0.08PRU
Right Ventricular Outflow Resistance	0.003PRU
Left Ventricular Outflow Resistance	0.006PRU
Left Ventricular Inflow Resistance	0.01PRU

Table 3.1: Normal Values of the Input Parameters in CVsim

puter models of the cardiovascular system, and a graphical approach for analyzing the regulation of the cardiac output based on the model [15].

Historically, there were several hemodynamic models designed mainly for teaching. In [2], Beeuwkes first used the analog computer in the hemodynamics teaching laboratory at Harvard Medical School. In 1978, Katz and his colleagues introduced a real time digital computer model for hemodynamics [18]. In his model, he simulated a Windkessel circulation, which is an over-simplified model in hemodynamic analysis. In 1982, Campbell et al. [4] developed a five component model for hemodynamics. In that model, he incorporated linear vascular elastances , resistances and time-varying ventricular capacitances proposed in [34].

In 1984, as part of the Athena project at M.I.T., Prof. Roger Mark proposed to design a cardiovascular model as a teaching tool for Quantitative Physiology and Cardiovascular Pathophysiology courses to provide students a quantitative understanding of the hemodynamics system. Based on Robert Sah's model shown in Figure 3-1, Timothy Davis implemented it the X window system in 1990 as his Master's project.

3.2 Hemodynamic Model

In analyzing the function of the cardiovascular system, it is convenient to represent the hydraulic properties of the various elements in terms of electrical circuit analogs. Table 3.2 lists the hydraulic variables and their electrical analogs.

Hydraulic Variable	Electrical Variable
Pressure (P)	Voltage (V)
Flow (J)	Current (i)
Volume (Q)	Charge (q)
Resistance R=$\frac{\Delta P}{J}$	Resistance R=$\frac{\Delta V}{i}$
Capacitance C=$\frac{\Delta Q}{\Delta P}$	Capacitance C=$\frac{\Delta q}{V}$

Table 3.2: Analog between Hydraulic System and Electrical System

First, let us focus on the peripheral circulation. We may characterize the large arterial vessels as a single lumped capacitance element, C_a. The microvasculature, which includes arterioles, capillaries and venules, can be characterized as a single resistance R_a. The main venous branches can be represented by a lumped venous capacitance C_v. In addition, we use a resistance R_v to simulate venous return. Then we get a simplified electrical circuit model for peripheral vasculature as shown in Figure 3-2. Actually, the venous resistance is distributed throughout the whole venous system. So a more accurate model of the venous system would be a distributed system for the capacitances and resistances.

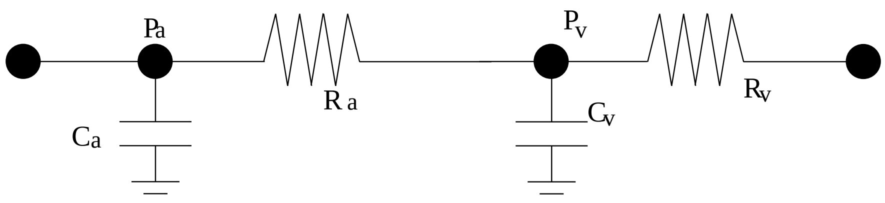

Figure 3-2: Electrical Circuit Model of Peripheral Vasculature

Now let us consider a model for the heart-lung unit. Then later on, we can integrate this unit with the peripheral system to form the model of the whole cardiovascular system. In [34], Sagawa et al. stated that if we examined a series of ventricular pressure-volume loops for a range of different preloads and afterloads, we would find that the end systolic points all lie on a straight line. The diastolic pressure volume relation may also be described by a straight line. Now we may use a variable capacitor to simulate the behavior of left ventricle. Considering the blood flow direction in the heart and the function of the cardiac values, we may add two diodes in the circuit. In addition, an output resistance is added to the left ventricle to simulate the impedance of the blood passing through the aortic valve. Then we get the electrical circuit model of the left heart as shown in Figure 3-3.

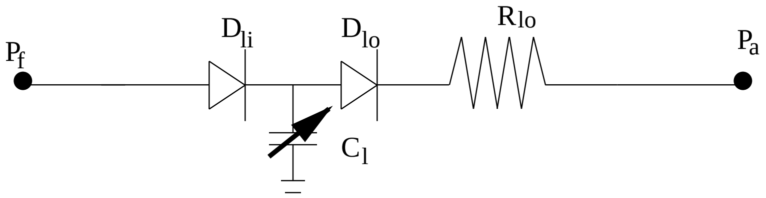

Figure 3-3: Electrical Circuit Model of Left Ventricle

The pulmonary circulation and the right ventricle can be modeled similarly. Combining the four functional units, we get a lumped parameter model shown in Figure 3-1. The normal values of the resistances and the capacitances in Figure 3-1 can be estimated by the relative resistance and volume in a normal person's cardiovascular system.

3.3 Baroreflex Model

In order to generate a more realistic simulation, the CVsim adds a baroreflex model to simulate the major cardiovascular feedback system. The primary goal of the central regulatory mechanisms is to maintain arterial blood pressure P_A within a very narrow range. The arterial baroreflex is the principal mediator of short-term control. The control system preserves P_A by feeding back transduced values of P_A. A feedback and control diagram is shown in Figure 3-4.

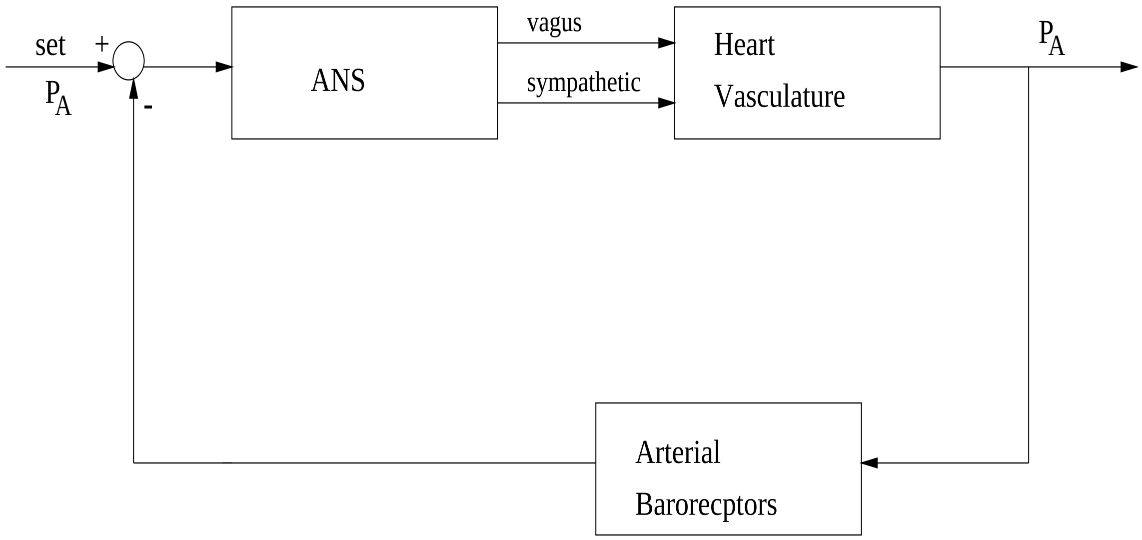

Figure 3-4: Feedback and Control Diagram of Baroreflex System. The autonomic nervous system (ANS) receives pressure difference then trigger the adjustment of the hemodynamic state.

In our research, since we always assume that we deal with the data when patients are in stable state, not in transient state, and thus we may turn off the baroreflex.

3.4 Some Example Cases

By using CVsim, we can simulate the hemodynamic state of different ICU patients:

- The primary effect of *mitral stenosis* is increased R_{li}. The compensatory effects of mitral stenosis can be simulated by increasing heart rate, increasing total blood volume, and increasing the contractility of the left and right ventricles.

- The primary effect of *vasodilation* is decreased R_a. The compensatory effects can be simulated by increasing heart rate, increasing total blood volume, and increasing the contractility of the left ventricle.
- The primary effect of *hypertension* is increased R_a. The compensatory effects can be simulated by increasing heart rate, decreasing venous zero-pressure volume, decreasing left ventricular diastolic capacitance, decreasing left ventricular systolic capacitance, decreasing venous zero-pressure volume, and increasing arterial resistance.
- The primary effect of *massive bleeding* is decreased total blood volume. The compensatory effects can be simulated by increasing heart rate, and decreasing left ventricular systolic capacitance.
- The primary effect of *septic shock* is decreased R_a. The compensatory effects can be simulated by increasing heart rate, increasing total blood volume, and increasing the contractility of the left ventricle.
- The primary effect of *tricuspid stenosis* is increased R_v. The compensatory effects can be simulated by increasing heart rate, and increasing total blood volume.

In the following chapters, we will develop a method to automatically identify the actual parameters of the model based on measurable data such as arterial blood pressure, central venous pressure, cardiac output, etc. Due to the limitation of the configuration of the circuit, there are some patient cases which are not possible to simulate, such as mitral regurgitation, tricuspid regurgitation, ventricular septal defect, etc.

Chapter 4

Reasoning Model

In this chapter, we will talk about the methodology used in our expert system. Traditionally, the first issue coming out of building an expert system is how to organize and represent knowledge. That is a typical scenario when people used to give qualitative results. In recent years, with the rapid development of mathematical tools, many researchers started to use pure or mainly pure numerical methods to try to get quantitative results. During the investigation of our current system, we tried unsuccessfully to use numerical methods, such as the homotopy method, to quantitatively solve the problem. Eventually, a new non-numerical method was proposed based on the analysis of medical decision making process.

4.1 A Typical Expert System

The process of building an expert system is often called knowledge engineering. It typically involves a special form of interaction between the knowledge engineer and experts in specific domains. The knowledge engineer "translates" the experts' knowledge, procedures, and strategies for problem solving into a computer program, as shown in Figure 4-1.

The ideal result will be a computer program successfully solving problems in the same manner as human experts. The heart of an expert system is the corpus of knowledge that accumulates during system building. In expert system design, the main role

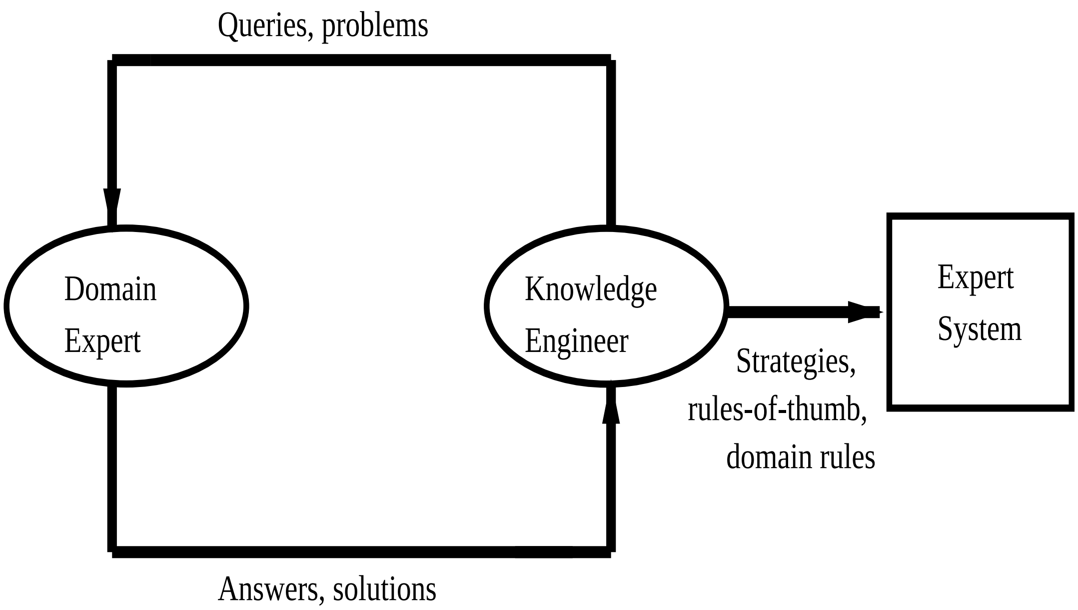

Figure 4-1: Knowledge Engineering, from [38]

players are the expert system, the domain expert, the AI consultants, the knowledge engineer, the expert system building tools, and the user. Their relationships to each other are shown in Figure 4-2.

The domain expert is a knowledgeable person with a reputation for producing right results in a specific domain. The knowledge engineer is a person with a background in AI, and is in charge of collaborating with domain expert and AI consultants. The AI consultants are respected persons in AI domain, who can always provide guidance to the knowledge engineer regarding general expert system questions. In my research, Prof. Roger Mark has been serving as the domain expert in cardiovascular diseases, Prof. Peter Szolovits has provided some interesting thoughts in searching for the methodology for the system as the AI consultant. I am the knowledge engineer. The system platform is the ANSI C programming language on a normal UNIX system.

4.1.1 Representation of Knowledge

The heart of an expert system is its corpus of knowledge, as we mentioned before. The three most widely used knowledge representation in current expert systems are rules, semantic nets, and frames.

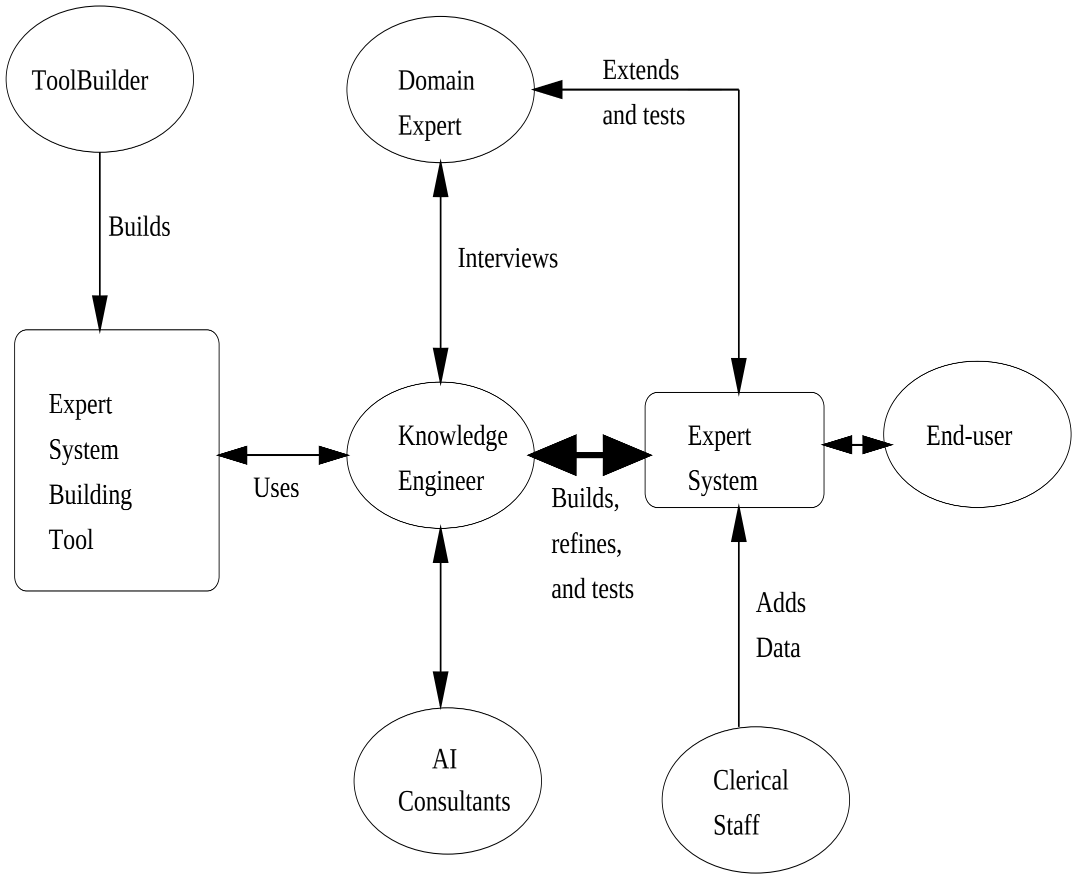

Figure 4-2: Roles in Expert System Building, adapted from [38]

- A **rule-based representation** uses the common statement *IF ... THEN ...*. The matching of rule IF portions to the facts can produce inference actions. Rules provide a natural way to describe problem-solving strategies. We can think about a scenario when we see a doctor and tell him what the problem is. For example, we might state:
  "I have abdominal pain ..., I have ..."
  The doctor will make the diagnosis based on our statements and some additional lab tests. How do doctors make their diagnosis? One way to think about is that they usually come up with some reasoning processes like:
  " ... because he has abdominal pain ... he may have stomach problems ..."
  Actually, the physicians' inference actions result from a rule such as:
  "*If* there is abdominal pain, *then* there is a possibility for the patient to have

stomach problem."

This reasoning technique is called *forward chaining*, because the system uses information on the left-hand side (*IF* part) to generate new information on the right side (*then* part). In the situation of a very large expert system with many rules, sometimes, *backward chaining* is more cost effective. With this inference method, the system starts with what it wants to verify and only executes rules that are relevant to proving it.

- A **semantic-net-based representation** uses a semantic net to represent knowledge based on a network structure. A semantic net consists of nodes connected by arcs that describe the relations between nodes.The *isa* and *has-part* relations are basic relations in a semantic net. Semantic nets are very useful to represent knowledge in domains using taxonomies [38].

- A **frame-based representation** uses a frame to represent knowledge. A frame is very much like a semantic net. In a frame, the topmost nodes represent general concepts, and the lower nodes represent some instances of the concepts. The concept at each node is defined as a collection of attributes and slots that are values of those attributes. Each slot can have several procedures attached to it. There are three commonly used procedures: *If-added, If-removed*, and *If-needed* procedures [38].

4.2 Problem Statement

In our present study, we used the CV simulator to generate "pseudo-clinical" test data which was representative of a variety of disease states. This approach simplified the task of designing the expert system, and also provided a quantitative method to evaluate the performance of our expert system. Since the actual input parameters of the CV model are known for each test case, they can be compared directly with those derived from the expert system.

The CV simulator is initialized by specifying each of the 23 model parameters.

The simulator then generates the resultant pressures, flows, and volumes at all sites in the CV system. The raw waveforms are then pre-processed by the "feature-detector" which derives a set of 13 clinically observable parameters (features) such as heart rate, mean arterial blood pressure, pulse pressure, mean central venous pressure, pulmonary arterial wedge pressure, cardiac output, etc. (Note that clinically non-observable parameters are not included in the feature set.) This feature set is used to characterize the physiologic data generated by the model. The diagram of generating and pre-processing data is shown in Figure 4-3.

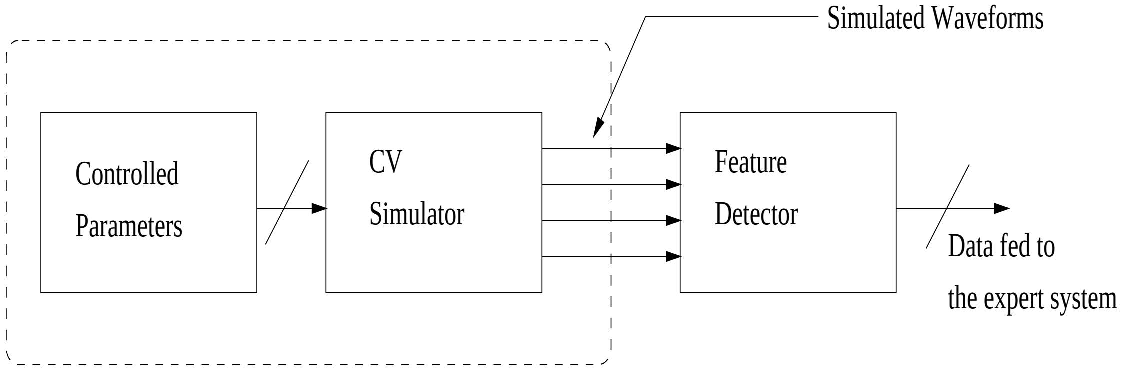

Figure 4-3: Diagram of Generating and Pre-processing Data

The input to the expert system is the 13 dimensional feature set from the patient or "pseudo-patient". Now the task of our expert system has been defined as how to derive a set of 23 control parameters for the CV model which result in output waveforms and features which match those of the patient.

4.3 A Pure Numerical Method for Quantitative Assessment

Since our task is to derive 23 control parameters for the CV model, we can think of the problem as a 23 dimensional search problem. This led us to think about applying some traditional pattern recognition methods to this problem.

In [40], in order to solve some non-linear algebraic equations, the homotopy method was used. When we started to look for a good numerical method to do this 23 dimensional search, we applied the homotopy method to our problem. In the following sections, I will start with an introduction to the homotopy method, then discuss how we applied this method to our problem, and some lessons learned from the work.

4.3.1 Introduction to the Homotopy Method

The concept of homotopy was originally introduced in topology. It is a very useful concept in categorizing different geometric shapes. Then this method was used in computer vision research, typically in object recognition. Morgan et al. in [26] first proposed to extend the homotopy concept into solving non-linear algebraic equations. In algebra, they define a homotopy as a schedule for transforming the start system (or equations) into the target system (equations). The homotopy process can be written as follows:

$$H(y,t) = (1-t)G(y) + tF(y) \tag{4.1}$$

where $G(y)$ is the starting system and $F(y)$ is the target system. Here y represents a set of variables. Suppose we know the solutions of $G(y)$ (we can always assume so because we can make up some algebraic equations with known roots) and we want to get the solutions of $F(y)$. Note that at $t = 0$, the solutions of $H(y,t) = 0$ are those of $G(y) = 0$, and at $t = 1$ the solutions of $H(y,t) = 0$ are those of $F(y) = 0$. The central idea of homotopy method is if we can track the changes of the solutions from $H(y,0)$ to $H(y,1)$, we will be able to solve $F(y)$, as shown in Figure 4-4.

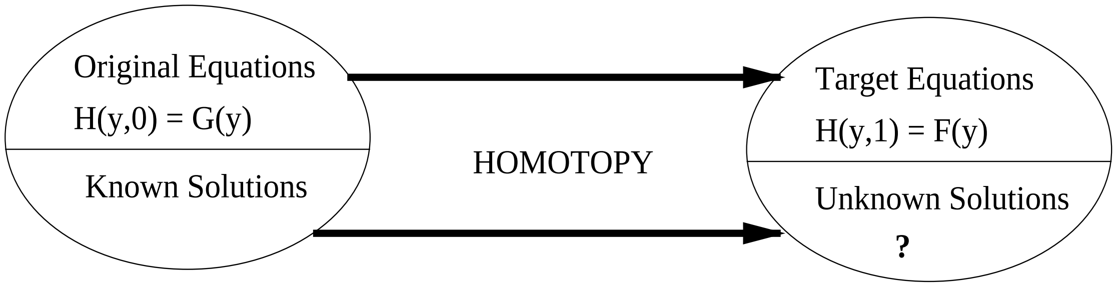

Figure 4-4: Homotopy Method for Solving Non-linear Equations

Using Eq. 4.1 to track a path from a known solution (y^0, t^0), we first predict the solution for $t = t^0 + \Delta t$ and then correct the prediction using Newton's method with t fixed. For small Δy and Δt, the Taylor series of expansion for H gives:

$$H(y + \Delta y, t + \Delta t) \cong H(y, t) + J_y \Delta y + J_t \Delta t \tag{4.2}$$

where J_y and J_t are the Jacobians of H with respect to y and t. In the prediction step, we have:

$$H(y^0, t^0) = 0 \tag{4.3}$$

and

$$H(y^0 + \Delta y, t^0 + \Delta t) = 0 \tag{4.4}$$

Then we can plug Eq. 4.3 and Eq. 4.4 into Eq. 4.2 and get:

$$\Delta y = -J_y^{-1} J_t \Delta t \tag{4.5}$$

Since Eq. 4.5 is only an approximate solution, we correct the solution at the new value of t by setting $\Delta t = 0$ to get a correction step:

$$\Delta y = -J_y^{-1} H(y, t) \tag{4.6}$$

We can repeat the correction step several times till we feel comfortable with the maximum error. Then we can incrementally increase t again, and go back to Eq. 4.2 and calculate the solutions under new t.

4.3.2 How to Apply the Homotopy Method to Our Problem

As we have discussed before, we can consider our problem as a 23 dimensional pattern search problem. By using the homotopy method, we can construct a system with known input parameters. For example, we can always start from the normal CV simulator parameters. What we know is the output feature of patients or pseudo-patients. We can incrementally make an effort to closely match the output features, and meanwhile, track the changes of the CVsim control parameters. Ideally, we will get the patient parameters setting at the end. The diagram is shown in Figure 4-5.

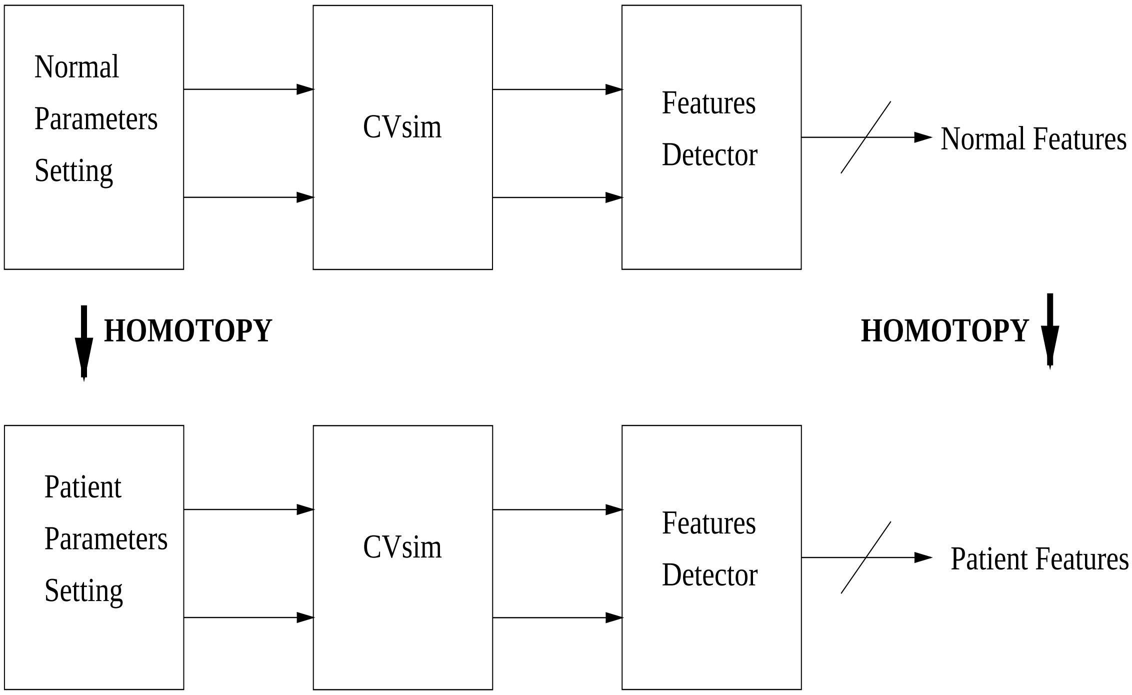

Figure 4-5: Diagram of Homotopy Method for Searching the Pattern of Patients

In our system, suppose $\vec{x} = (x_1, x_2, \cdots, x_N)$ is a vector representing the input systemic parameters of CVsim and $\vec{y} = (y_1, y_2, \cdots, y_N)$ is a vector representing the output features. The procedures of applying homotopy method is as follows:

1. Set a new $\vec{y^d}$, anticipate the new $(\Delta x_1, \Delta x_2, \cdots, \Delta x_N))$ by solving the linear equations:

$$\begin{cases} \Delta y_1 \approx \frac{\partial y_1}{\partial x_1}|_{\vec{x}}\Delta x_1 + \frac{\partial y_1}{\partial x_2}|_{\vec{x}}\Delta x_2 + \cdots + \frac{\partial y_1}{\partial x_N}|_{\vec{x}}\Delta x_N \\ \Delta y_2 \approx \frac{\partial y_2}{\partial x_1}|_{\vec{x}}\Delta x_1 + \frac{\partial y_2}{\partial x_2}|_{\vec{x}}\Delta x_2 + \cdots + \frac{\partial y_2}{\partial x_N}|_{\vec{x}}\Delta x_N \\ \cdots \\ \Delta y_M \approx \frac{\partial y_M}{\partial x_1}|_{\vec{x}}\Delta x_1 + \frac{\partial y_M}{\partial x_2}|_{\vec{x}}\Delta x_2 + \cdots + \frac{\partial y_M}{\partial x_N}|_{\vec{x}}\Delta x_N \end{cases} \tag{4.7}$$

2. Let $\vec{x'} = \vec{x} + \vec{\Delta x}$.

3. Run CVsim and get $\vec{y'} = (y_1', y_2', \cdots, y_N')$.

4. Set $\Delta\vec{y'} = \vec{y'} - \vec{y^d}$.

5. Refine new $\vec{x'}$ by solving the linear equations:

$$\begin{cases} \Delta y_1' \approx \frac{\partial y_1}{\partial x_1}|_{\vec{x'}} \Delta x_1' + \frac{\partial y_1}{\partial x_2}|_{\vec{x'}} \Delta x_2' + \cdots + \frac{\partial y_1}{\partial x_N}|_{\vec{x'}} \Delta x_N' \\ \Delta y_2' \approx \frac{\partial y_2}{\partial x_1}|_{\vec{x'}} \Delta x_1' + \frac{\partial y_2}{\partial x_2}|_{\vec{x'}} \Delta x_2' + \cdots + \frac{\partial y_2}{\partial x_N}|_{\vec{x'}} \Delta x_N' \\ \cdots \\ \Delta y_M' \approx \frac{\partial y_M}{\partial x_1}|_{\vec{x'}} \Delta x_1' + \frac{\partial y_M}{\partial x_2}|_{\vec{x'}} \Delta x_2' + \cdots + \frac{\partial y_M}{\partial x_N}|_{\vec{x'}} \Delta x_N' \end{cases} \tag{4.8}$$

6. Repeat step 2 to step 5 until satisfied.

7. Check whether $\vec{y^d}$ has reached the target $\vec{y^T}$, if yes, then stop; otherwise go back to step 1.

One of the problems is that we do not have the $\frac{\partial y_i}{\partial x_j}$ function directly available. We need to run CVsim to get a huge 23 dimensional look-up table to describe the partial derivative function.

4.3.3 Some Lessons Learned

We implemented the homotopy method for solving the problem. Unfortunately, it did not work as well as we expected. We have learned some lessons from it. There are several concerns with using a pure numerical method on the model-based reasoning system:

- The method works very well if only 1 or 2 systemic parameters have been changed. This problem seems a traditional AI problem (*System only works for the simple cases*). But the issue here is not that simple. The basic idea of the homotopy method is more or less like a gradient method. What is the disadvantage of using a gradient method to solve some non-linear problems? As we all remember, the major concern is the local minimum vs. global minimum. Of course, we can try to add some techniques, such as simulated annealing, to try to avoid the local minimum. Stimulated annealing allows the search to take some uphill steps to escape from the local minimum. Theoretically, system will escape from a local minimum, maybe after many steps. In realistic situations, those techniques are not applicable because of time issues and stability issues.

- By using this numerical method, we must ask ourselves a question: how much can we trust the CVsim? The reason why this comes out as an issue is that in order to do the gradient based analysis, we need to get a huge partial derivative look-up table. This look-up table can only be generated from running the CVsim.
- The numerical method is always trying to minimize the total error. However, at the same time, it ignores two main problems:
  - For each reasoning (moving) process, it does not follow some common sense strategies. In other words, the method does not consider the real physiological status. This is why it may give out some ridiculous answers.
  - As we will find out in the next chapter, the decrease of the total error of the output features between two systems does not necessarily mean that the input parameters are getting closer.

4.4 Design of the Current System

All is not lost with the failure of the homotopy method. It reminded us to look for some less numerical and more heuristic methods to deal with the reasoning process. Our goal is to produce quantitative interpretation or at least semi-quantitative interpretation. Why is the quantitative analysis much more difficult than the qualitative analysis? If I make an analogy to the problem of searching for an apple in a room, quantitative analysis means that we need to be able to figure out not just the direction of the move, but also how much the move should be. Intuitively, it is almost impossible to reach the apple by a single step move (we all have to move and then look, then move, ...), unless we are very lucky. So in order to perform quantitative analysis, we can get some feeling about the design strategies from grabbing an apple in a room. That is, the method must be iterative, must be able to tell me something at each step, which is related to how far away toward our target after each move. In other words, our expert system must be able to run CVsim iteratively, and search for the best match. Then the diagram of the current system was put forward in [41], as shown in Figure 4-6.

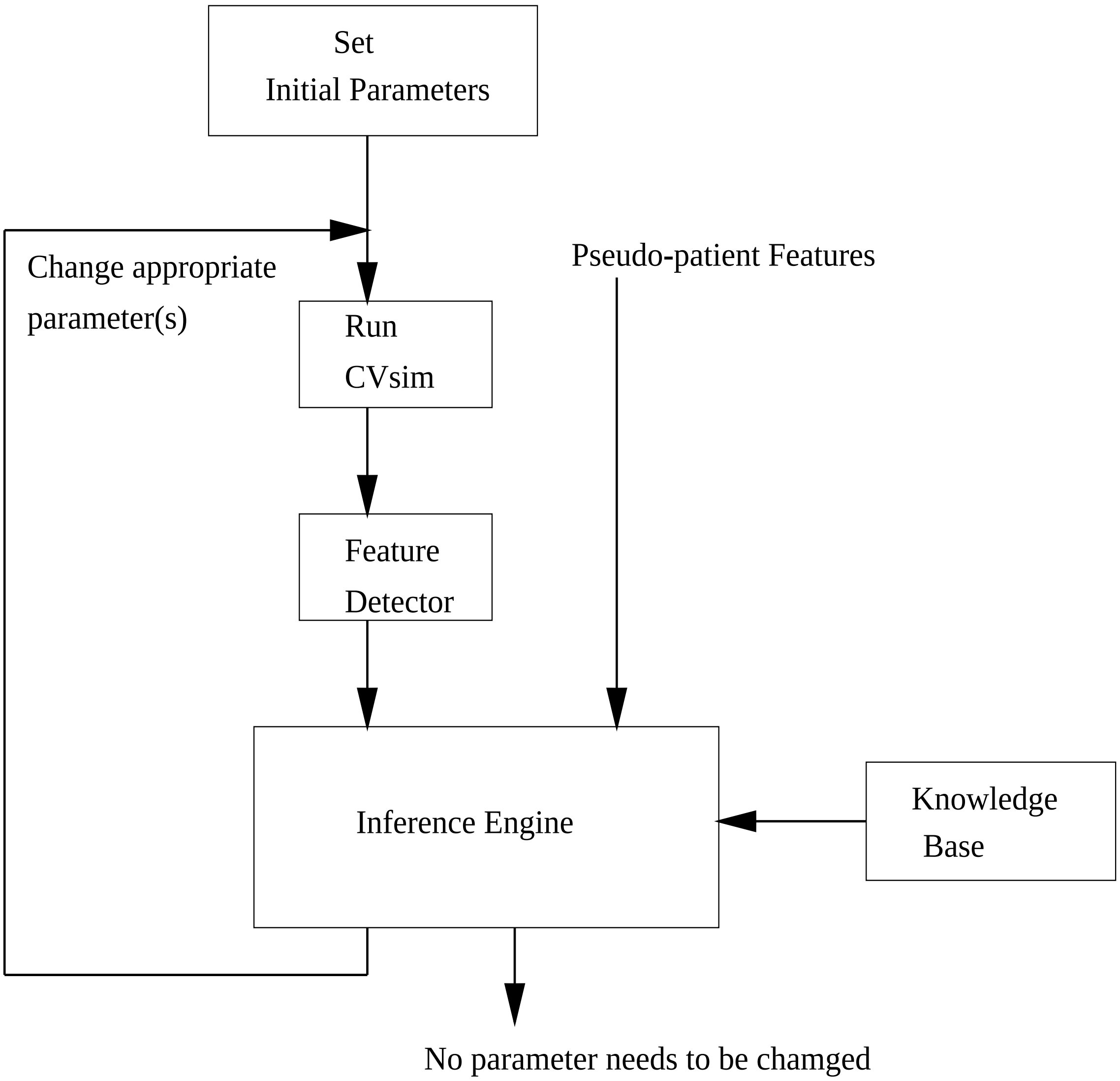

Figure 4-6: Diagram of Our Expert System

We first establish an initial value for each of the systemic input parameters. Then we run CVsim. From the output variables of CVsim, such as mean arterial blood pressure, mean central venous pressure, systemic arterial pulse pressure, we use the inference engine to determine which input parameter(s) need to be changed in order to decrease the differences between the outputs from CVsim and the pseudo-patient. Again, after adjusting the input systemic parameters, we re-run CVsim. This iteration process is continued until we feel satisfied with the difference between the outputs from CVsim and the pseudo-patient. The comparison between them is made by using the following error function:

$$\mathbf{E}_i = \frac{|Pseudopatient\ Output\ Value - CVsim\ Output\ Value|}{Pseudopatient\ Output\ Value}$$

For example, the error function of the mean CVP (Central Venous Pressure) is defined as follows:

$$\mathbf{E}_{cvp} = \frac{|Pseudopatient\ mean\ CVP - CVsim\ mean\ CVP|}{Pseudopatient\ mean\ CVP\ Value}$$

In fact, if we look at the Figure 4-6 and compare with the strategy we developed in the homotopy method, we may notice that we actually inherited the concept of homotopy. The only major difference is that we use heuristic methods to design the inference engine instead of pure mathematical methods.

4.4.1 Initial Values of Systemic Parameters

In our system, we always start with calculating some computable systemic parameters, including:

1. Heart Rate (beats/min).
2. Systemic Microvascular Resistance R_a (mmHg*sec/ml):

$$R_a = \frac{Mean\ SABP - Mean\ CVP}{Cardiac\ Output/60}$$

ABP: Systemic Arterial Blood Pressure; CVP: Central Venous Pressure.

3. Pulmonary Microvascular Resistance R_{pa} (mmHg*sec/ml):

$$R_{pa} = \frac{Mean\ PAP - Mean\ PCWP}{Cardiac\ Output/60}$$

PAP: Pulmonary Arterial Pressure; PCWP: Pulmonary Capillary Wedge Pressure.

4. Systemic Arterial Capacitance C_a (ml/mmHg):

$$C_a \cong \frac{Stroke\ Volume}{Pulse\ Pressure}$$

Note that we do not calculate some resistances that look calculable, such as *Right Ventricular Outflow Resistance*. Theoretically, we can use

$$R_{ro} = \frac{Mean\ RVP - Mean\ PAP\ (when\ RVP > PAP)}{Cardiac\ Output/60}$$

to estimate R_{ro} (PAP: Pulmonary Arterial Pressure, RVP: Right Ventricular Pressure). The problem with the above formula is that the normal value of this resistance is very small, equal to 0.003PRU. Assume that this person's cardiac output is normal (about 5000cc/min), then the mean pressure difference during systole between pulmonary arterial pressure and right ventricular pressure will be:

$$0.003(mmHg * sec/ml) * \frac{5000(ml/min)}{60(sec/min)} = 0.25mmHg$$

We can not assume that in real clinical areas, 0.25mmHg is detectable. So unless there is severe valvular stenosis symptom, I assume that these resistances are normal. As a matter of fact, physicians usually make the decision about pulmonary valve stenosis based on the instantaneous pressure gradient between the right ventricle and the pulmonary artery.

There are also several parameters which do not change based on our assumptions. They are:

1. Transthoracic Pressure: The changes of the transthoracic pressure can change the relative filling pressure. Ignoring the changes of transthoracic pressure, we eliminate the effects of the respiratory system. So our system can not be used for the patients on ventilators.

2. All Zero-pressure Volumes: The changes of the zero-pressure volumes can be compensated by changing the total blood volume, because from the hemodynamic point of view, we are concerned with the effective blood volume (=total blood volume - the summation of zero-pressure volumes).

3. Venous Capacitance: Clinically, incremental venous capacitance does not change in most of the patient cases.

All of the remaining input CVsim parameters are initialized to normal default values.

4.4.2 Flow Chart of the Reasoning Process

In our system, medical knowledge is represented as rules. We have two flow charts in our system:

- Figure 4-7, Figure 4-8 and Figure 4-9 show the reasoning process for patients with no valvular disease.

- Figure 4-10, Figure 4-11 and Figure 4-12 show the reasoning process for analyzing patients with possible mitral stenosis and/or tricuspid stenosis diseases.

To determine which flow chart the system uses for each patient, we start to run our expert system while asking some questions, such as:

- Is there any diastolic murmur?

- Is there any sign of mitral stenosis from the echocardiogram?

The detection of tricuspid stenosis is much easier than the detection of the mitral stenosis, since we have the measurement of right ventricular pressure. We can make the decision based on the instantaneous pressure gradient between the central venous pressure and the right ventricular pressure.

4.4.2.1 Flow Chart for Non-valvular Disease Patients

Figure 4-7 shows the top level of the reasoning process. It always starts from checking the difference of the cardiac output between the pseudo-patient and the CVsim. The comparison gives three possible results:

- *High* means the cardiac output of the CVsim is higher than that of the pseudo-patient.
- *Equivalent* means the cardiac output of the CVsim is within the same range (5% for cardiac output) with that of the pseudo-patient.
- *Low* means the cardiac output of the CVsim is lower than that of the pseudo-patient.

Since we assume there is no valvular disease in this flow chart, there is no action to change the input and output flow resistances. The possible changing systemic parameters are: total blood volume, left ventricular systolic capacitance, right ventricular systolic capacitance, left ventricular diastolic capacitance, right ventricular diastolic capacitance, systemic arterial capacitance and pulmonary arterial capacitance. In Figure 4-7, *tune up* means the situation of the pseudo-patient's heart is better than that of the CVsim. In this case, we need to tune up the systemic parameters of CVsim. *Tune down* is the opposite situation of *tune up*. Figure 4-8 continues the reasoning process of tune up or down. In Figure 4-7, *adjust* means we need to change both the systolic capacitance and the diastolic capacitance of left/right ventricle. Figure 4-9 continues the reasoning process of adjustment.

Finally, if all the measurable parameters of CVsim are considered to be equivalent with that of the pseudo-patient, the iteration should *stop*. It means the current CVsim's systemic parameters can be used to simulate the status of the pseudo-patient.

4.4.2.2 Flow Chart for Patients with Possible Mitral Stenosis and Tricuspid Stenosis Diseases

From the echocardiogram, we may find some evidence of mitral stenosis and/or tricuspid stenosis. The possible changing systemic parameters are: total blood volume, left ventricular systolic capacitance, right ventricular systolic capacitance, left ventricular diastolic capacitance or inflow resistance, right ventricular diastolic capacitance or inflow resistance, systemic arterial capacitance and pulmonary arterial capacitance. Note that in the case of mitral stenosis, pulmonary wedge pressure is not equal to the left ventricular end diastolic pressure. Figure 4-11 continues the reasoning process of tune up or down. Figure 4-12 continues the reasoning process of adjustment. If there are signs of mitral stenosis, we change the value of left ventricular inflow resistance instead of the value of the left ventricular diastolic capacitance. If there are signs of tricuspid stenosis, we change the value of right ventricular inflow resistance instead of the value of the right ventricular diastolic capacitance.

4.4.3 Some Heuristic Components in Our Expert System

In designing an expert system, there are a number of other important factors that we need to think about besides which parameter(s) need to be changed. A very important factor in designing the search control process is how to define the step size for each legitimate move. An easy approach is to set a very small fixed step size. The problem with the small fixed step size is obvious: it may take us a very long time to get to the target point from the original point. So in our system, we decided to use variable step sizes instead. The way that we set/change step size during the search control process is described as follows:

- We set initial unit step sizes, which usually tend to be conservative, when we start the search control process.
- The step size is proportional to the difference between the outputs of the CVsim and the pseudo-patient. For example, we change the systemic arterial capaci-

tance based on the difference of the pulse pressure of the CVsim and the pseudo-patient. The larger the difference is, the larger is the step size.

$$
\begin{aligned}
&Step\ Size\ of\ Systemic\ Arterial\ Capacitance = \\
&\quad |Pseudopatient\ Pulse\ Pressure - CVsim\ Pulse\ Pressure| * \\
&\quad Unit\ Step\ Size
\end{aligned}
$$

- We set an increase unit step size and a decrease unit step size separately for each parameter. The reason for this strategy is that we want to avoid oscillation. The disadvantage of defining only one unit step size for both decrease and increase is shown in Figure 4-13.
- If the new proposed change for a parameter is exactly in the same direction as the last change, we increase the appropriate unit step size by 50%. This is what we called "Reward" strategy, as shown in Figure 4-14.
- If the new proposed change for a parameter is in the opposite direction with the last change, we will go back to the place where we were, then decrease the appropriate unit step size by 50%. This is what we called "Punish" strategy. Figure 4-15 shows how it works.

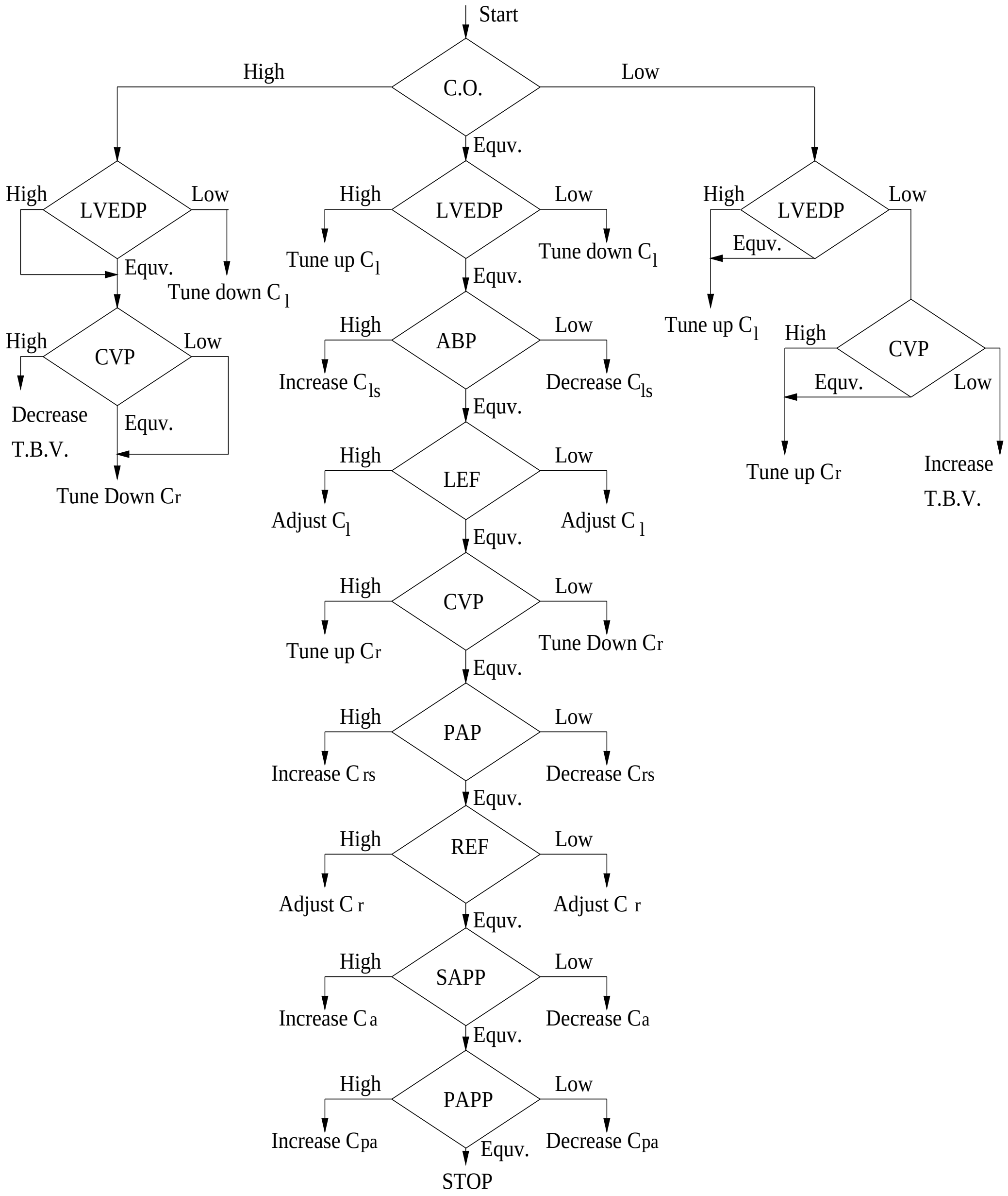

Figure 4-7: Top Level Flow Chart of Reasoning Process for Non-valvular Disease Patients. C.O.=Cardiac Output; LVEDP=Left Ventricular End-Diastolic Pressure; ABP=Mean Systemic Arterial Blood Pressure; LEF=Left Ejection Fraction; CVP=Mean Central Venous Pressure; PAP=Mean Pulmonary Arterial Pressure; REF=Right Ejection Fraction; SAPP=Systemic Arterial Pulse Pressure; PAPP=Pulmonary Arterial Pulse Pressure; Equv.=Equivalent. Continued with Figure 4-8 and Figure 4-9.

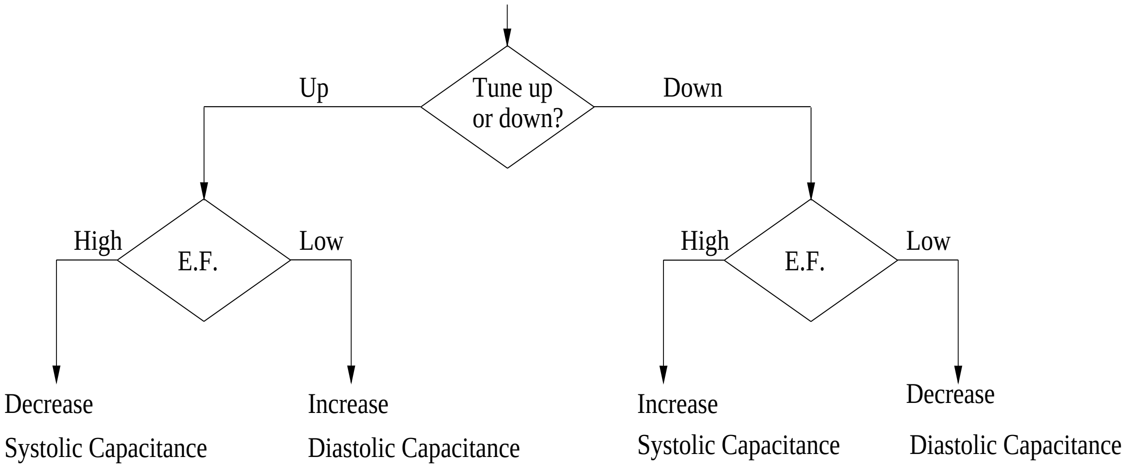

Figure 4-8: Tune Up/down Reasoning Process for Non-valvular Disease Patients. Continued for Figure 4-7 when we need to tune or down the ventricular functions.

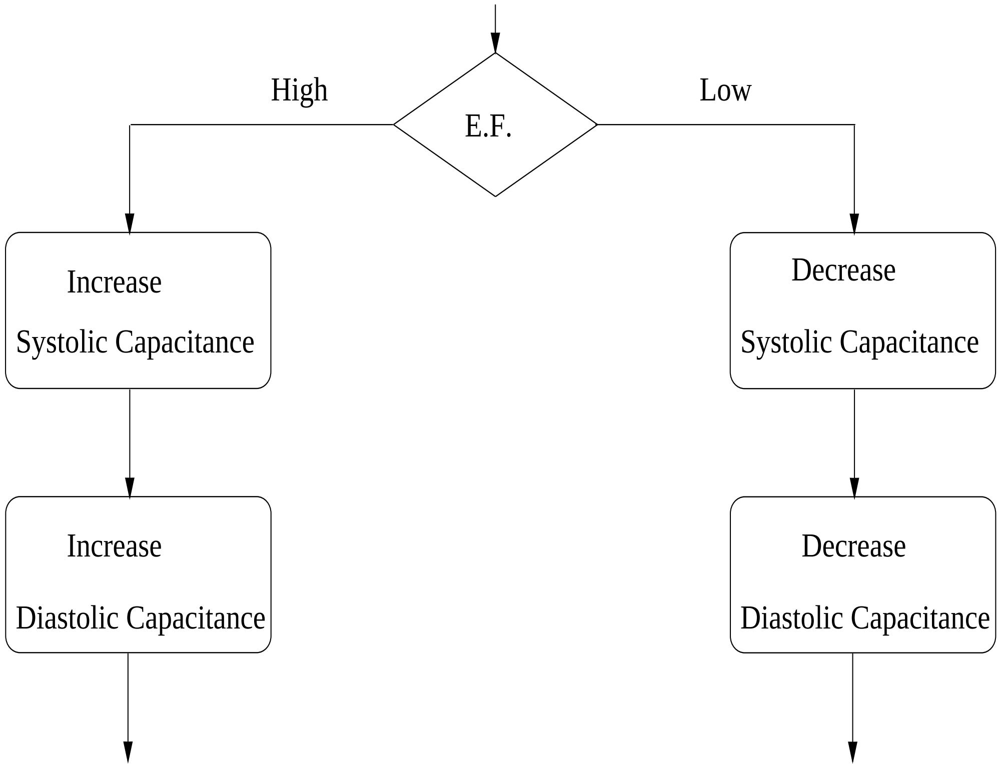

Figure 4-9: Adjust Reasoning Process for Non-valvular Disease Patients. Continued for Figure 4-7 when we need to adjust the ventricular capacitances based on ejection fraction.

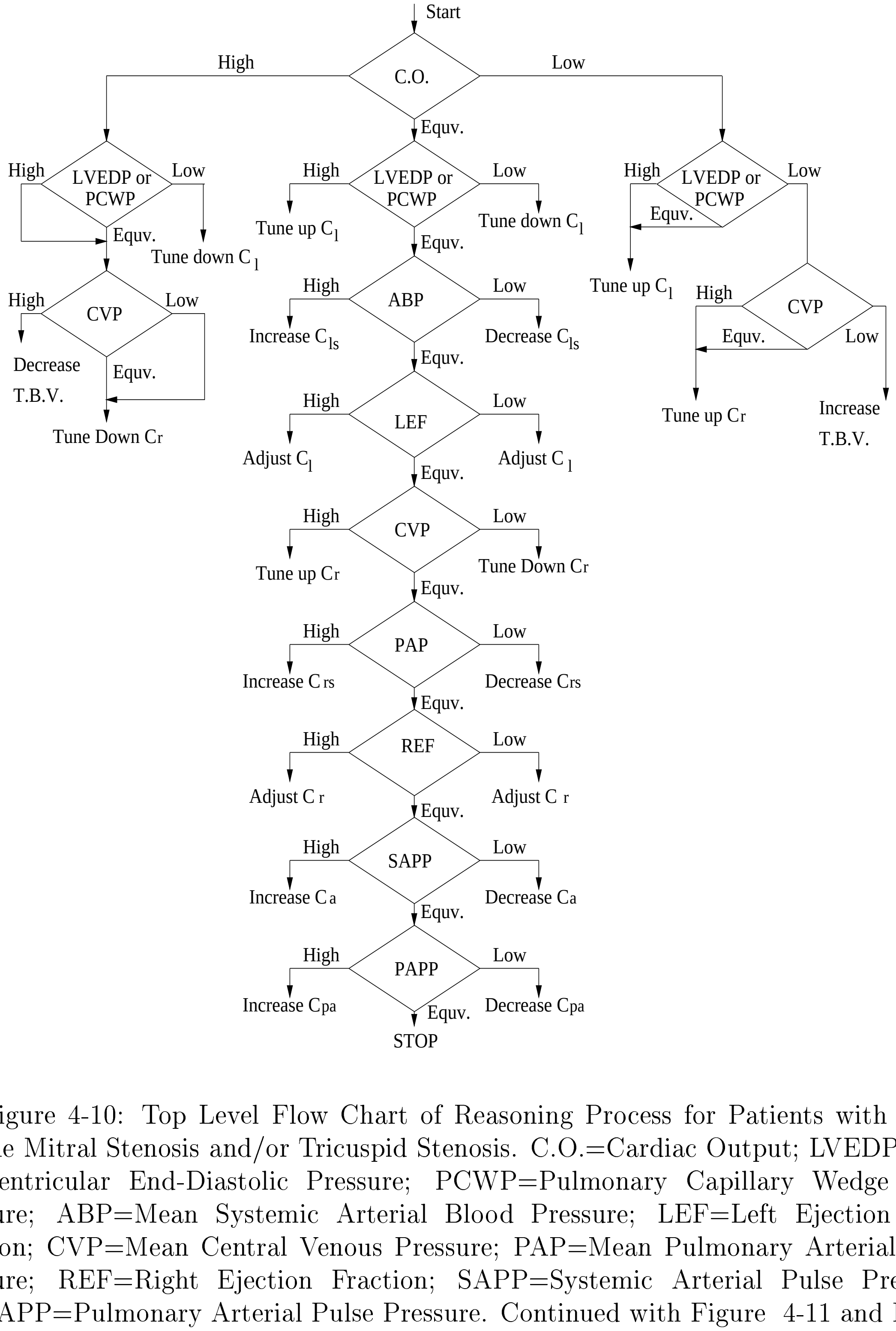

Figure 4-10: Top Level Flow Chart of Reasoning Process for Patients with Possible Mitral Stenosis and/or Tricuspid Stenosis. C.O.=Cardiac Output; LVEDP=Left Ventricular End-Diastolic Pressure; PCWP=Pulmonary Capillary Wedge Pressure; ABP=Mean Systemic Arterial Blood Pressure; LEF=Left Ejection Fraction; CVP=Mean Central Venous Pressure; PAP=Mean Pulmonary Arterial Pressure; REF=Right Ejection Fraction; SAPP=Systemic Arterial Pulse Pressure; PAPP=Pulmonary Arterial Pulse Pressure. Continued with Figure 4-11 and Figure 4-12.

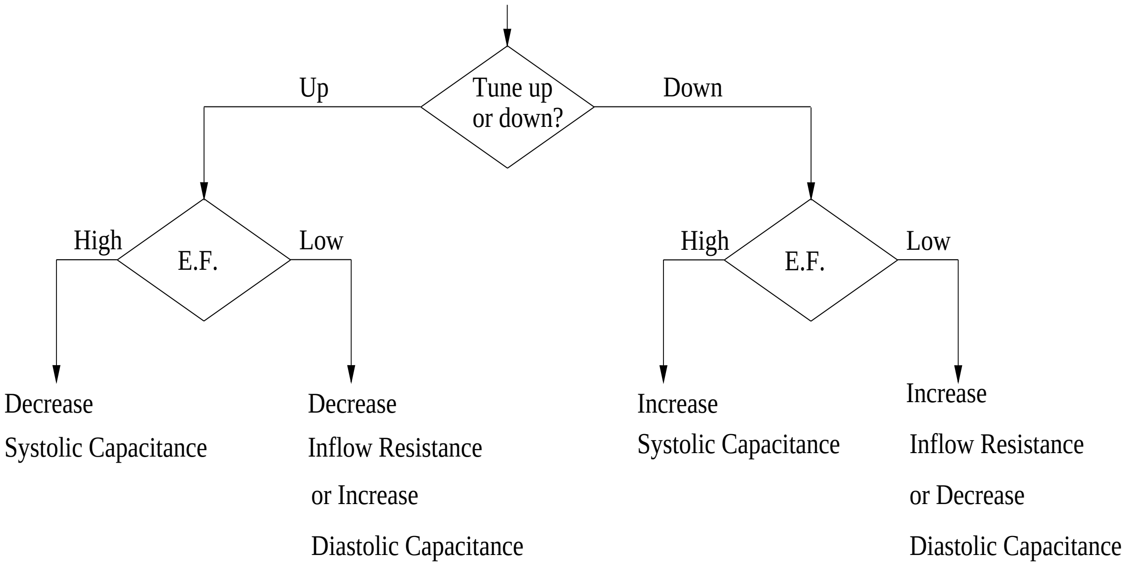

Figure 4-11: Tune Up/down Reasoning Process for Patients with Possible Mitral Stenosis and/or Tricuspid Stenosis. Continued for Figure 4-10 when we need to tune or down the ventricular functions.

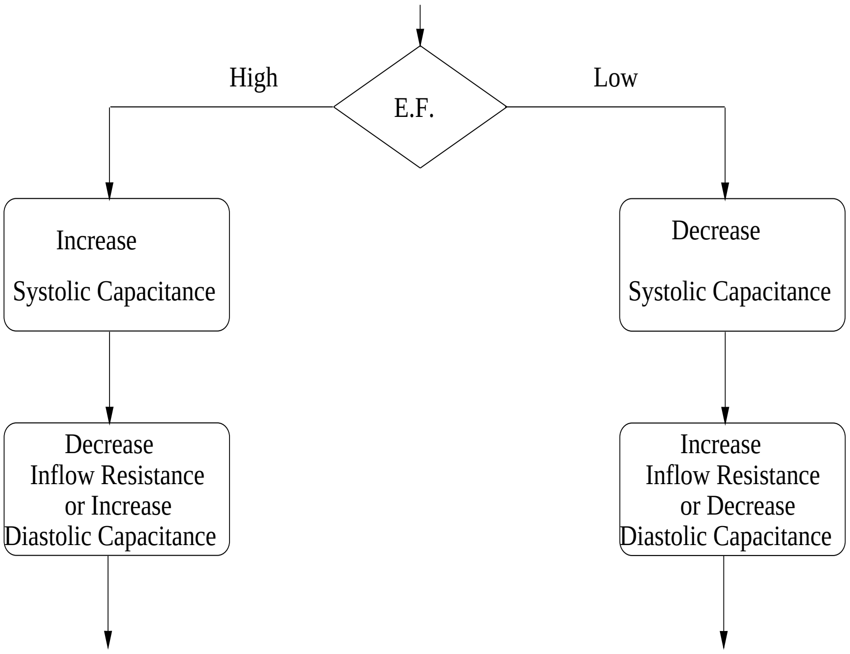

Figure 4-12: Adjust Reasoning Process for Patients with Possible Mitral Stenosis and/or Tricuspid Stenosis. Continued for Figure 4-10 when we need to adjust the ventricular capacitances based on ejection fraction.

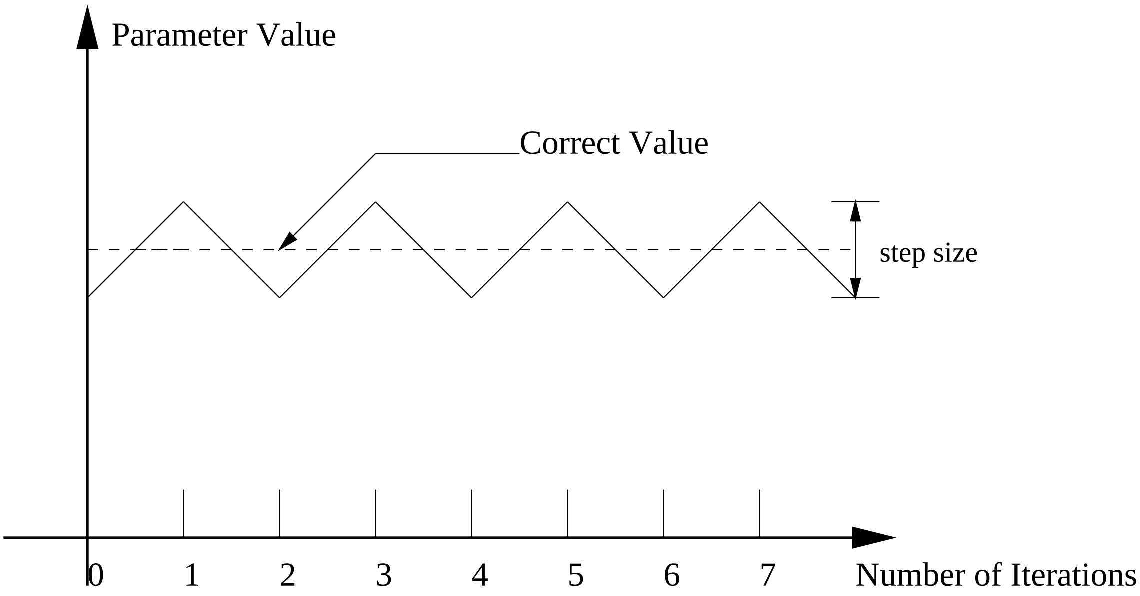

Figure 4-13: Why do we need two unit step sizes for each parameter? As we can see, we may overshoot sometimes. If there is only one unit step size for both increase and decrease, that may cause the system to oscillate.

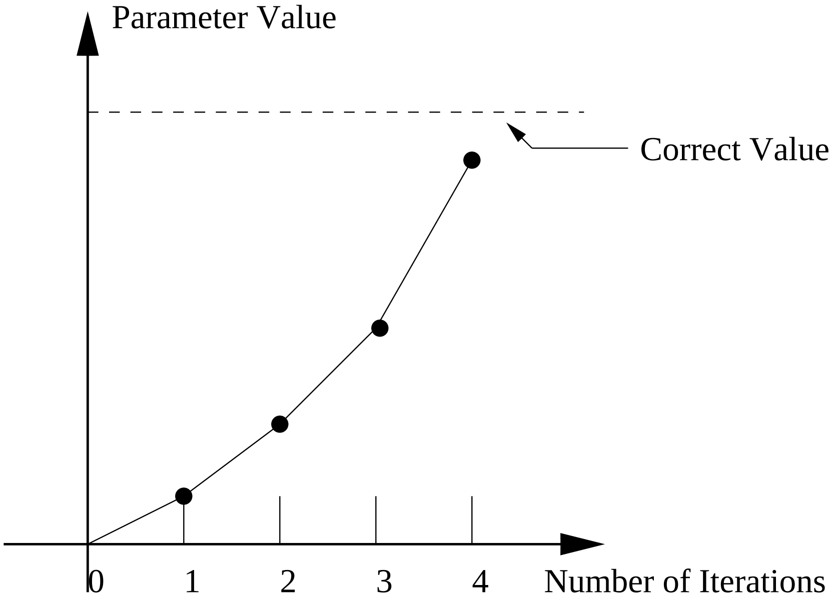

Figure 4-14: Reward Strategy. If the proposed change is in the same direction as the last change, we tend to think that we are still quite far away from our target, we need to be a little bit more aggressive in order to save iteration times.

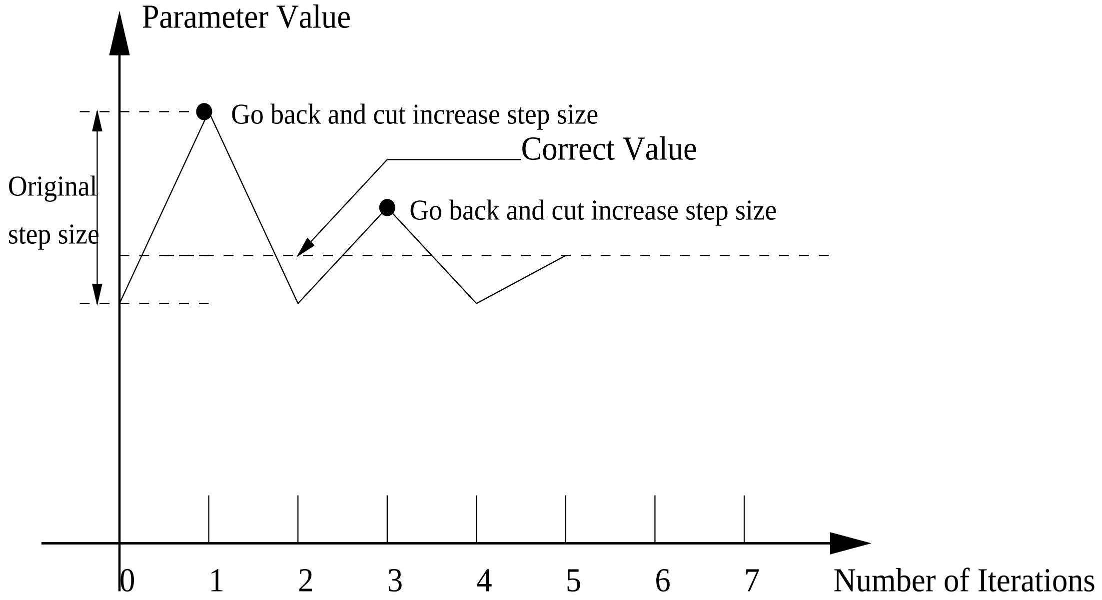

Figure 4-15: Punish Strategy. It is another strategy to avoid oscillation.

Chapter 5

Development and Evaluation of the System

Evaluating an expert system is very difficult because sometimes, there may be no formal way to prove a given answer is correct or acceptable or the best possible. The evaluation process usually involves consulting other experts (not the one where we had most of the domain knowledge) in the domain. Ultimately, user acceptance can make or kill an expert system. Obviously, users will not accept an expert system with poor performance; neither will they accept a high performance expert system that gives confusing answers, or is hard to learn how to use, or takes a long time to get results.

A number of evaluations of medical expert systems have been put forward in [19][22][25], most of them were focusing on evaluation of the qualitative analysis. In these evaluation processes, the results from expert systems are presented to human experts, and they are asked to fill a sheet for the evaluation for each case. In [22], reviewers were given the following 5 choices:

- **Correct** means the result is a reasonable accounting of the findings.
- **Possible** means the result is a possible interpretation of the findings, but not the best.

- **Partly Correct** means the result is mostly reasonable, but has some minor mistakes.

- **Wrong** means the result is wrong.

- **Seriously Wrong** means the result is wrong and dangerous.

It is even more difficult to evaluate the performance of our expert system, because our system provides quantitative results. An ideal evaluation of our program would begin by identifying a large number of actual patient cases that occur in the cardiovascular domain covered by this program. In this approach to evaluation, the actual patient data play two roles. First, they tell us how frequently various conditions arise, and therefore how important the program's performance is on such cases. Second, they provide us a natural variety of distinct cases against which to test our program.

Performing such an evaluation is quite difficult because of the need for a large number of cases. In addition, we believe that relatively small variations in the pseudo-patient case would not yield significantly better or worse performance by our program. Therefore, instead of testing on cases derived from actual patient data, we have evaluated our approach by simulating a varied set of clinically important and distinguishable cases that span the domain of expertise of the program. These cases can then illustrate the strengths and weaknesses of the program, although we cannot expect them to yield a statistical estimate of how frequently it would do well or poorly with situations abstracted from actual cases. Of course, such illustrations are exactly what we need in a formative evaluation, whose major purpose is to identify areas in which additional work will be needed.

While we were developing the reasoning and knowledge base of our system, we used a learning set of 16 simulated cases for both testing and refining the algorithms. In the next section, we will present and discuss the test results of our expert system on these cases.

5.1 Results and Discussion

All of the relevant tables and figures can be found in Appendix A. In this section, we will analyze the results and discuss potential improvements for our system. For each case, we will present two tables:

- Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters
- Comparison of Output Variables of Pseudo-Patient with those of Simulator

Two figures will also be provided for each case:

- Total Output Error Function vs. Number of Iterations: The total output error function is defined as:

$$E_{OUTPUT} = \sqrt{E_1^2 + \cdots + E_i^2 + \cdots + E_{13}^2}$$

$$\mathbf{E}_i = \frac{|Pseudopatient\ Output\ Value - CVsim\ Output\ Value|}{Pseudopatient\ Value}$$

- Total Input Error Function vs. Number of Iterations: The total input error function is defined as:

$$E_{INPUT} = \sqrt{E_1^2 + \cdots + E_i^2 + \cdots + E_{17}^2}$$

$$\mathbf{E}_i = \frac{|Pseudopatient\ Parameter\ Value - Estimated\ Parameter\ Value|}{Pseudopatient\ Parameter\ Value}$$

Note that we have 7 volume related input parameters for CVsim, but since we can only characterize the system state based on the total effective blood volume (=total blood volume - summation of all zero-pressure volumes), only the input error of the total effective blood volume is taken into account for calculating the total input error.

In addition, it is also very important to conduct a qualitative evaluation of the performance of the system. The domain expert (Dr. Roger Mark) who had helped to design the system was asked to participate in the qualitative evaluation. The system's estimated input parameters and CVsim's output variables were presented to Dr. Mark, and he was asked to provide a corresponding clinical diagnosis. The expert system's performance was then rated using the following scale:

- **Excellent** means that:
  1. The diagnosis implied by the expert system's estimated parameters is consistent with the original diagnosis modeled by the test case.
  2. The algorithm converges.
  3. The final total input error is less than 20%.
- **Acceptable** means that:
  1. The diagnosis implied by the expert system's estimated parameters is consistent with the original diagnosis modeled by the test case.
  2. The algorithm does not converge and/or the (final)[1] total input error is greater than 20%.
- **Unsatisfactory** means that the diagnosis implied by the expert system's estimated parameters is not consistent with the original diagnosis modeled by the test case, and would probably lead to an erroneous diagnosis in clinical areas.

5.1.1 Cases without Compensatory Effects

We start our discussion with the testing results on the cases without compensatory effects. These cases represent relatively simple physiological changes with respect to normal people.

[1]If the algorithm does not converge, then there is no *final* total input error.

5.1.1.1 Blood Loss

This case represents a patient with blood loss. The status of a blood loss patient was simulated on CVsim by decreasing total blood volume. After running about 4 minutes (8 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.1 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.2 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference of their normal values. Figure A-2 shows the output error function for each iteration. Since we know how we set the input parameters of CVsim, we also printed out how the input error function behaves for each iteration in Figure A-1. The final total input error is 7.4%, and the final total output error is 12.0%. In this case, the algorithm converges very well. The rating of this case given by the reviewer is *excellent*.

By comparing Figure A-2 and Figure A-1, we may notice that the drop of the output error function does not necessarily correspond to the drop of the input error function. For example, from the first iteration to the second iteration, there is a huge drop in the output error function; however, the input error function increases. This may be part of the reason why the homotopy method failed to solve this problem, because the basis of the homotopy method is how to achieve a smaller output error function for each step. In our system, we use total error function to determine whether we should stop the iteration, but we do not use the output error function to determine whether the move for each iteration is legitimate or not.

5.1.1.2 Arteriolar Dilation

This case represents a patient with arteriolar dilation, and was simulated on CVsim by decreasing peripheral resistance. After running about 2 minutes (4 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.3 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.4 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-4 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-3. The final total input error is 1.3%, and the final total output error is 1.6%. In this case, the algorithm converges very well. The rating of this case given by the reviewer is *excellent*.

5.1.1.3 Left Ventricular Systolic Dysfunction

This case represents a patient with left ventricular systolic dysfunction. The status of this patient was simulated on CVsim by increasing the left ventricular systolic capacitance. After running about 3 minutes (7 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.5 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.6 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-6 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-5. The final total input error is 3.5%, and the final total output error is 2.8%. In this case, the algorithm converges very well. The rating of this case given by the reviewer is *excellent*.

5.1.1.4 Arteriolar and Venous Dilation

This case represents a patient with arteriolar and venous dilation. The case was simulated on CVsim by increasing venous zero-pressure volume, and decreasing peripheral resistance. After running about 4 minutes (9 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.7 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.8 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-8 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-7. The final total input error is 13%, and the final total output error is 5.5%. In this case, the algorithm converges very well. The rating of this case given by the reviewer is *excellent*.

5.1.2 Cases with Compensatory Effects

The following cases represent some pathological cases with compensatory effects. In these cases, primary changes are caused by the disease directly, and the compensatory effects are caused by various neural and humoral compensatory mechanisms.

5.1.2.1 Hypertension

This case represents a patient with hypertension. The primary effect was simulated on CVsim by increasing peripheral resistance. The compensatory effects were increased heart rate, decreased venous zero-pressure volume, increased left ventricular contractility, decreased left ventricular diastolic compliance, and decreased arterial compliance. After running about 10 minutes (24 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.9 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.10 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-10 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-9. The final total input error is 10.0%, and the final total output error is 9.7%. In this case, the algorithm converges very well. The rating of this case given by the reviewer is *excellent*.

5.1.2.2 Right Ventricular Infarction

This case represents a patient with right ventricular infarction. The primary effect was simulated by increasing right ventricular systolic capacitance and decreasing right ventricular diastolic capacitance. The compensatory effects were increased heart rate and increased peripheral resistance. The program did not stop by itself because it stayed in a local minimum state. Table A.11 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system during the 24th iteration (system takes about 10 minutes to run 24 iterations). Table A.12 shows the comparison of the output variables of the pseudo-patient to those of the simulator during the 24th iteration, with reference to their normal values. Figure A-12 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-11. The algorithm did not converge in this case. The rating of this case given by the reviewer is *acceptable*.

From Table A.12, we can see that just by comparing the output variables during the 24th iteration, the simulator output is very much like the pseudo-patient's output. It means that if we loosen our comparison boundary (with the sacrifice of the precision of results), the program could have stopped by itself. From Table A.11, we notice that the problem is that our system tended to "think" that the right ventricular systolic function is worse than it should be, and the right ventricular diastolic function is better than it should be. The system gets stuck in this local minimum state. In order to avoid the local minimum state, we may apply a similar strategy with simulated annealing [30], which allows us to impose a probabilistic value for each branch in the decision tree to increase the uncertainty during each iteration. For example, if the mean pulmonary arterial pressure of the CVsim is higher than that of the pseudo-patient, the current algorithm will always increase the right ventricular systolic capacitance. By imposing a probabilistic value (for example: 80%) to this rule, the algorithm may execute the rule by 80% chance or continue the search process to check the right ventricular ejection fraction by 20% chance.

5.1.2.3 Hypertrophic Cardiomyopathy

This case represents a patient with hypertrophic cardiomyopathy without outlet obstruction. The primary effect was simulated by decreasing left ventricular systolic capacitance, decreasing left ventricular diastolic capacitance, and decreasing right ventricular diastolic capacitance. The compensatory effects were increased heart rate, increased total blood volume, decreased venous zero-pressure volume, and decreased peripheral resistance. After running about 30 minutes (74 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.13 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.14 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-14 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-13. The final total input error is 14.6%, and the final total output error is 13.6%. In this case, the algorithm converges well, but not very fast. The rating of this case given by the reviewer is *excellent*.

From Figure A-14, we can see some interesting behavior of the output error function during the 5th iteration to 11th iteration. There is a huge rise of the input error function during the 7th iteration, and a huge drop during the 8th iteration, then another huge rise during the 9th iteration. That is when our *punish* strategy started to work. During the 7th iteration, the reasoning process increased the total blood volume by 2983.42ml. Then during the 8th iteration, the reasoning process *tended* to decrease the total blood volume. However, before it really decreased the total blood volume, it *looked* backward and *noticed* that the problem was last time, the reasoning process increased the total blood volume way too much. Then the program went back to the position of the 6th iteration, and increased the total blood volume by 2237.18ml. Again, we see that the changes of the total output error function does *not* necessarily correspond to the changes of the total input error function.

5.1.2.4 Pulmonary Hypertension

This case represents a patient who has pulmonary hypertension. The primary effect was simulated by increasing pulmonary resistance. The compensatory effects were decreased right ventricular systolic capacitance, increased heart rate, increased total blood volume, and decreased pulmonary capacitance. The program did not stop by itself, because it stayed in a local minimum state. Table A.15 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system during the 24th iteration. Table A.16 shows the comparison of the output variables of the pseudo-patient to those of simulator during the 24th iteration, with reference to their normal values. Figure A-16 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-15. The rating of this case given by the reviewer is *acceptable*.

The major problems of the result are too much effective blood volume and no change of pulmonary arterial capacitance. The problem of the algorithm is because of the negative effect of our *punish* strategy. During the 66th iteration, the algorithm decreased the left ventricular systolic capacitance. However, on the 67th iteration, the algorithm tended to increase the the left ventricular systolic capacitance. Then it noticed that the problem may be because last time the left ventricular systolic capacitance had been decreased too much. The algorithm decreased the unit step size. Then it was still overshoot. Then according to our *punish* strategy, the unit step size needed to be decreased again. Finally, it stayed in that position which was always trying to decrease this capacitance, but never succeeded. To solve this problem, we may impose a probabilistic value for each branch in the decision tree to increase the uncertainty during each iteration.

5.1.2.5 Large AV Fistula

This case represents a patient with large AV fistula. The primary effect was simulated by decreasing peripheral resistance. The compensatory effects were increased heart rate, increased total blood volume, decreased left ventricular systolic capacitance, decreased right ventricular systolic capacitance, decreased venous zero-pressure volume, and decreased pulmonary resistance. After running about 10 minutes (23 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.17 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.18 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-18 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-17. The final total input error is 55.7%, and the final total output error is 6.1%. In this case, the algorithm converges well. The rating of this case given by the reviewer is *acceptable*.

The major difference between the results and the set-up parameters is that our expert system tended to add extra blood volume and worsen the diastolic functions of both ventricles a bit more. From Table A.18, we can see that the output observable parameters are very close.

5.1.2.6 Dilated Cardiomyopathy and Bi-ventricular Congestive Heart Failure

This case represents a patient with dilated cardiomyopathy and bi-ventricular congestive heart failure. The primary effect was simulated by increasing left ventricular systolic capacitance, increasing right ventricular systolic capacitance, decreasing left ventricular diastolic capacitance, increasing left ventricular zero-pressure volume, and increasing right ventricular zero-pressure volume. The compensatory effects were increased heart rate, increased total blood volume, decreased arterial capacitance, increased right ventricular diastolic capacitance, decreased pulmonary arterial capacitance, decreased venous zero-pressure volume, and increased pulmonary resistance. After running about 35 minutes (82 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.19 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with the reference to their normal values. Table A.20 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-20 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-19. The final total input error is 52.0%, and the final total output error is 16.8%. In this case, the algorithm converges well, but not fast enough. From Figure A-20, we can find the similar behavior of the output error function as discussed in hypertrophic cardiomyopathy case. The rating of this case given by the reviewer is *acceptable*.

5.1.2.7 Acute Myocardial Infarction with Bi-ventricular Heart Failure

This case represents a patient with an acute myocardial infarction and bi-ventricular heart failure. The primary effect was simulated by increasing left ventricular systolic capacitance, increasing right ventricular systolic capacitance, and decreasing left ventricular diastolic capacitance. The compensatory effects were increased heart rate, decreased arterial capacitance, increased right ventricular diastolic capacitance, decreased pulmonary arterial capacitance, decreased venous zero-pressure volume, increased peripheral resistance, and increased right ventricular zero-pressure volume. After running about 33 minutes (77 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.21 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.22 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-22 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-21. The final total input error is 14.6%, and the final total output error is 10.1%. The rating of this case given by the reviewer is *excellent*. Again, from Figure A-22 and Figure A-21, we may notice that sometimes, even if there is a huge rise of the output error, there may not be a rise of the input error.

5.1.2.8 Septic Shock

This case represents a patient in septic shock. The primary effect was simulated by decreasing peripheral resistance, increasing venous zero-pressure volume, and increasing left ventricular systolic capacitance. The compensatory effect was increased heart rate. After running about 22 minutes (51 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.23 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with the reference to their normal values. Table A.24 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-24 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-23. The final total input error is 14.2%, and the final total output error is 10.7%. In this case, the algorithm converges well. The rating of this case given by the reviewer is *excellent*.

5.1.2.9 Pulmonary Hypertension and Right Heart Failure

This case represents a patient with pulmonary hypertension and right heart failure (cor pulmonale). The primary effect was simulated by increasing pulmonary resistance, and increasing right ventricular systolic capacitance. The compensatory effects were increased heart rate, increased total blood volume, decreased peripheral arterial capacitance, decreased venous zero-pressure volume, increased peripheral resistance. After running about 23 minutes (50 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.25 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with the reference to their normal values. Table A.26 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-26 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-25. The final total input error is 6.6%, and the final total output error is 12.8%. In this case, the algorithm converges very well. From Figure A-26, we can see again the *punish* strategy has worked. The rating of this case given by the reviewer is *excellent*.

5.1.2.10 Mitral Stenosis I

This case represents a patient with mitral stenosis. The primary effect was simulated by increasing left ventricular inflow resistance. The compensatory effects were increased heart rate, increased total blood volume, decreased right ventricular systolic capacitance and increased pulmonary resistance. In this case study, we intentionally indicated to the algorithm that there is *no* evidence of mitral stenosis and tried to see what results we would get. After running about 13 minutes (30 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.27 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.28 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-28 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-27. The final total input error is 120.3%, and the final total output error is 7.3%. The rating of this case given by the reviewer is *unsatisfactory.*

The major difference of our results with set-up parameters are the left ventricular inflow resistance and the left ventricular diastolic capacitance. However, clinically, it is also very hard to determine the mitral stenosis case without knowing this person's left ventricular pressure. Our expert system explained the problem as a left ventricular filling problem, which is correct because with a very high left ventricular diastolic pressure, we usually turn our concern to this person's diastolic problems.

5.1.2.11 Mitral Stenosis II

This case represents a patient with mitral stenosis. The primary effect was simulated by increasing left ventricular inflow resistance. The compensatory effects were increased heart rate, increased total blood volume, decreased right ventricular systolic capacitance and increased pulmonary resistance. In this case study, we indicated to the algorithm that there is evidence of mitral stenosis. After running about 11 minutes (27 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.29 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.30 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-30 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-29. The final total input error is 16.5%, and the final total output error is 11.1%. The algorithm converges well in this case. The rating of this case given by the reviewer is *excellent.*

The current algorithm for analyzing the valvular disease is based on the assumption that if there is any inflow valvular problem, we assume that the diastolic capacitance is normal.

5.1.2.12 Tricuspid Stenosis

This case represents a patient with tricuspid stenosis. The primary effect was simulated by increasing right ventricular inflow resistance. The compensatory effects were increased heart rate, and increased total blood volume. After running about 10 minutes (23 iterations) on a Sun SPARC-10 station, the iteration stops. Table A.31 shows the comparison of the input parameters of the pseudo-patient to those estimated by the expert system, with reference to their normal values. Table A.32 shows the comparison of the output variables of the pseudo-patient to those of the simulator, with reference to their normal values. Figure A-32 shows the output error function for each iteration. The input error function for each iteration is shown in Figure A-31. The final total input error is 4.3%, and the final total output error is 4.6%. The algorithm converges very well in this case. The rating of this case given by the reviewer is *excellent*.

5.2 Summary of the Testing

We tested our expert system on 16 typical cases, in which 11 cases were rated as excellent, 4 cases were rated as acceptable and 1 case was rated as unsatisfactory. For the unsatisfactory case, the reason for the wrong diagnosis is lack of knowledge. This evaluation provides us with a reasonable picture of the diagnostic performance of our expert system. The number of cases in the evaluation was too small to measure the performance relative to physicians. In general, the more complicated the case is, will the longer time the system take to figure out the problem. There is no serious error made by the expert system, but there are some problems, such as time issues, and lack of convergence. They imply that further refinement of the reasoning process is needed rather than the fundamental changes to the reasoning mechanisms and methodology. Currently, we are pleased with the results and believe that the system can be improved.

Chapter 6

Conclusion

In this thesis, we have developed an expert system for quantitatively or semi-quantitatively analyzing ICU hemodynamic data. We have designed a new iterative approach, which uses a cardiovascular simulator as a model. The application of this methodology can also be applied outside the area of medicine to solve system identification problems. In our system, clinical rules and other available test results (such as an echocardiogram) have been used to guide the direction of the iteration in our system. The results of our case studies suggested that this approach is promising for solving other clinical problems. There are also some unsolved problems in our system:

- How can we analyze aortic stenosis and pulmonary stenosis? For pulmonary stenosis, we can make the quantitative analysis based on the pressure gradient between right ventricular pressure and pulmonary arterial pressure. For aortic stenosis, it will be very difficult, because left ventricular pressure monitoring is not a routine procedure in ICU areas. Without left ventricular pressure data, aortic stenosis can possibly be detected by echocardiogram. However, we need to be careful to use the quantitative description provided by the echocardiogram.

- How to further increase the efficiency of the system? Or how to decrease the iteration times? Setting more aggressive unit step sizes may be one way, but that may also increase the instability of the system.

Due to the limitation of the cardiovascular simulator that we used in our research, our system can not be used to analyze some cases such as mitral regurgitation.

6.1 Practical Consideration

Of course, eventually, we hope to use our system in ICU areas by feeding the system with real patient data. Then there will be several major problems that need to be solved:

- Time issue: what is an acceptable time period for the system to converge on an answer?

- As we have shown in two of our cases, sometimes the system may just stay in a local minimum state and may iterate forever. Obviously, we cannot count on the end-users to stop the program in that case. Then how to avoid this situation? We may set a maximum iteration time, once the total iteration time reaches the maximum iteration time, we can just stop the program and choose the input parameters which correspond to the minimum output error as our estimation.

- We need to pre-process the real ICU data to get rid of noise, artifacts, etc.

6.2 Future Related Work

There are a number of improvements that can be made to our current system:

- For realizing the real time analysis, we need to have some temporal rules instead of static rules. Actually, real time analysis may be easier than figuring out the problem for the first time, because we do not expect a dramatic change from one minute to another minute. It means that the distance from the "target" to the "original" state is closer.

- During the period of writing this thesis, I came up with an idea: we can set up some typical cases as the starting point for our system. For example, if we roughly know what kind of problems that the patient may have, for instance, tricuspid stenosis, then we can set a tricuspid case as our original parameters instead of always starting from normal values. It will certainly save lots of iterations and decrease the possibility of getting wrong answers. An intuitive explanation to this idea is that again, the distance from the "target" to the "original" state is closer.

- Since clinically ECG is very easy to record and get, we can connect our system with ECG analysis that may help our search process. For example, ECG diagnoses for myocardial infarction (MI), hypertrophy, etc., can be used in establishing the initial diagnosis.

- We can also try other ways to do the reasoning process. For example, in our current system, we use a flow chart to describe the reasoning process. It is part of the reason that most of the answers seem to *favor* left ventricular problem. A change that we can make is that for each iteration, we find the biggest difference among all observable data and think about how to decrease this difference instead of always starting from matching cardiac output, pulmonary wedge pressure, etc. Or we can impose a probabilistic value to each decision branch. In that way, we may increase the possibility of escaping from a local minimum state.

- We noticed that part of the reason why sometimes the system seems to get stuck, is because it is hard for the system to figure out that, "*Hey, I have been here before*". How to detect and describe this situation? We may define several domains in the multi-dimensional space to categorize the states of the system. Once we find out the system is back to a previous state, we can apply another reward strategy to increase the unit step size in this situation, so that the system can possibly get out from the local minimum state.

- In the long term, we hope to apply these methods to actual patient data. More summative evaluation methods will then need to be devised.

- Another problem we noticed is that the algorithm always tried to minimize the output errors for only the next step. That means our program lacks a global view of the problem. Just like playing the Chess game, sometimes, we may have to take a sacrifice in order to win the game eventually. Currently, our algorithm is not smart enough to sacrifice the output errors in certain iterations to eventually decrease the final total output error. How to make our system smarter? We may need to apply some general guidelines for each case, so that the system may seem to know which parameter(s) will need to be focused on globally.

6.3 Summary

In this project, we have developed an iterative method for quantitatively or semi-quantitatively analyzing ICU hemodynamic data. The key idea embodied in our method is that with the help of a model, it is possible to solve a multi-dimensional search problem by iteratively running the model, that is, *test and see*. The other qualitative description may help the search process. As concluded from the above discussion, we still have a long way to go before we can fully, accurately and efficiently automate the interpretation process. However, we believe that we have developed a successful prototype of the final system that will guide us towards our objective. Just as the IBM Deep Blue computer has won the Chess game, we are also optimistic that our system can be improved for practical use in ICU areas in the foreseeable future.

Appendix A

Case Study Results for Testing Expert System

In this appendix, we will present the case study results for testing our expert syetm. For each case, two tables:

- Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters
- Comparison of Output Variables of Pseudo-Patient with those of Simulator

and two figures:

- Total Output Error Function vs. Number of Iterations
- Total Input Error Function vs. Number of Iterations

will be presented.

A.1 Cases Without Compensatory Effects

A.1.1 Blood Loss

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	72	72	0%
Total Blood Volume (ml)	5000	**4500**[1]	**4528.78**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	0.4	0%
LVdiast	10.0	10	10	0%
Arterial	1.6	1.6	**1.69**	5.6%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	**1.18**	1.7%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500.0	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**675**	**703.78**	4.2%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.00	**0.99**	1%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.079**	1.3%
Total Number of Iterations	8	Total Input Error		7.4%

Table A.1: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Blood Loss Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	63.2	64.5	2.1%
Maximum Systemic Arterial Pressure (mmHg)	113.2	76.1	77.1	1.3%
Minimum Systemic Arterial Pressure (mmHg)	75.6	50.5	52.0	3.0%
Mean Central Venous Pressure (mmHg)	6.6	3.2	3.4	6.3%
Cardiac Output (ml/min)	5321	3634	3739	2.9%
Heart Rate (beats/min)	72	72	72	0%
Mean Right Ventricular Pressure (mmHg)	10.6	6.0	6.2	3.3%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	9.9	10.2	3.0%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	13.9	14.3	2.9%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	6.1	6.3	3.3%
Pulmonary Wedge Pressure (mmHg)	9.2	5.0	5.3	6%
Left Heart Ejection Fraction (%)	55	52	52	0%
Right Heart Ejection Fraction (%)	62	58	59	1.7%
Total Number of Iterations	8	Total Output Error		12.0%

Table A.2: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Blood Loss Case

[1] The items in bold type are different from the normal values.

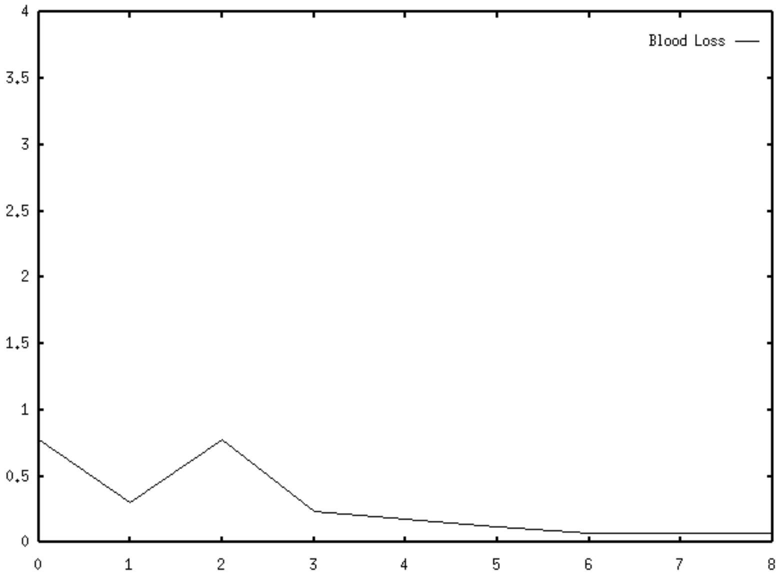

Figure A-1: Input Error Function vs. Number of Iterations in Blood Loss Case

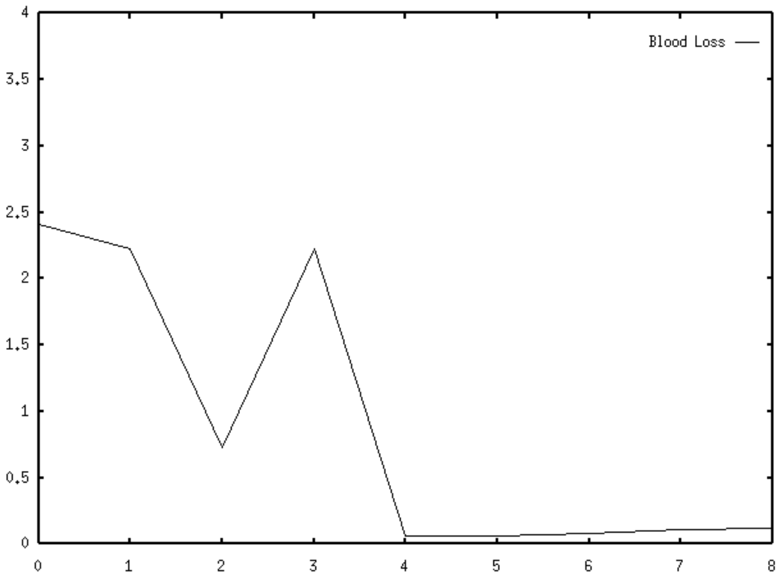

Figure A-2: Output Error Function vs. Number of Iterations in Blood Loss Case

A.1.2 Arteriolar Dilation

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	72	72	0%
Total Blood Volume (ml)	5000	5000	5000	0%
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	0.4	0%
LVdiast	10.0	10	10	0%
Arterial	1.6	1.6	1.6	0%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	1.2	0%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	1175	1175	0%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.5**	**0.50**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.079**	1.3%
Total Number of Iterations	4	Total Input Error		1.3%

Table A.3: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Arteriolar Dilation Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	55.4	55.2	0.4%
Maximum Systemic Arterial Pressure (mmHg)	113.2	75.4	75.0	0.5%
Minimum Systemic Arterial Pressure (mmHg)	75.6	37.0	36.6	1.1%
Mean Central Venous Pressure (mmHg)	6.6	7.3	7.3	0%
Cardiac Output (ml/min)	5321	5830	5838	0.1%
Heart Rate (beats/min)	72	72	72	0%
Mean Right Ventricular Pressure (mmHg)	10.6	11.2	11.2	0%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	16.2	16.2	0%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	22.6	22.6	0%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	10.2	10.1	1.0%
Pulmonary Wedge Pressure (mmHg)	9.2	8.5	8.5	0%
Left Heart Ejection Fraction (%)	55	64	64	0%
Right Heart Ejection Fraction (%)	62	64	64	0%
Total Number of Iterations	4	Total Output Error		1.6%

Table A.4: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Arteriolar Dilation Case

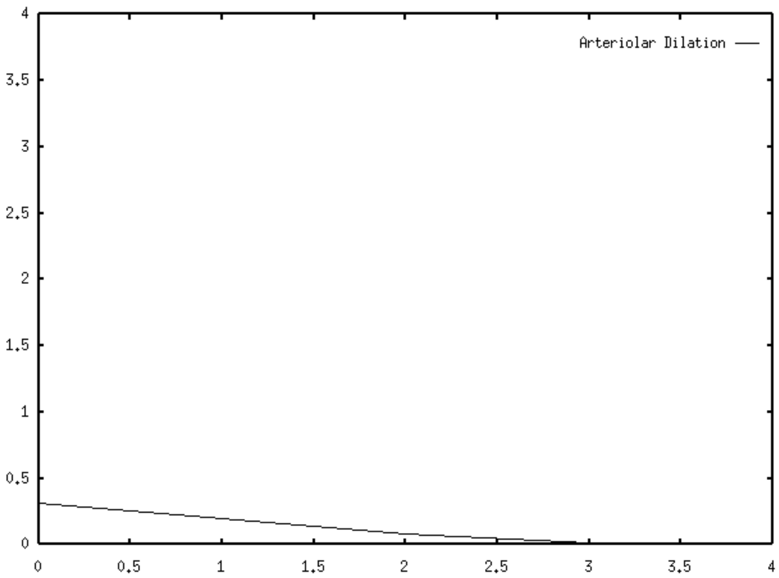

Figure A-3: Input Error Function vs. Number of Iterations in Arteriolar Dilation Case

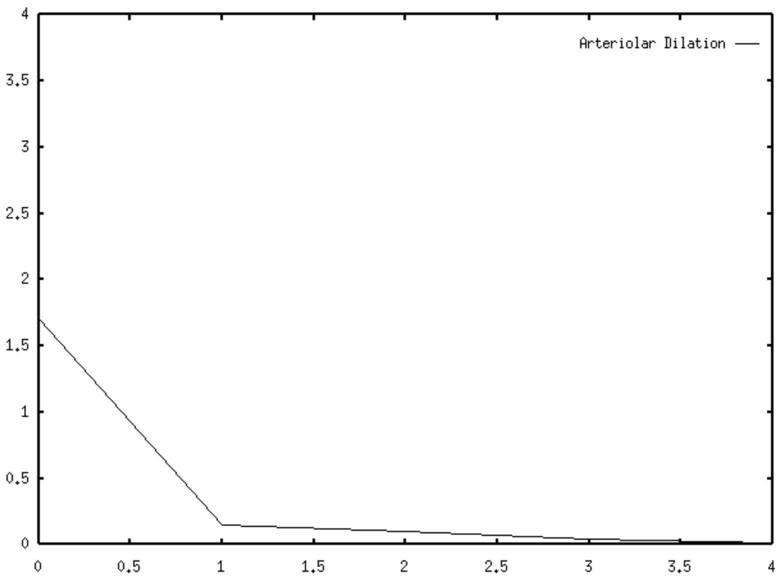

Figure A-4: Output Error Function vs. Number of Iterations in Arteriolar Dilation Case

A.1.3 Left Ventricular Systolic Dysfunction

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	72	72	0%
Total Blood Volume (ml)	5000	5000	5000	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**1**	**0.99**	1%
LVdiast	10.0	10	10	0%
Arterial	1.6	1.6	**1.64**	2.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	1.2	0%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	1175	1175	0%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.0	**0.98**	2%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.079**	1.3%
Total Number of Iterations	7	Total Input Error		3.5%

Table A.5: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Left Ventricular Systolic Dysfunction Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	81.0	80.3	0.9%
Maximum Systemic Arterial Pressure (mmHg)	113.2	96.6	95.6	1.0%
Minimum Systemic Arterial Pressure (mmHg)	75.6	65.2	64.9	0.5%
Mean Central Venous Pressure (mmHg)	6.6	5.8	5.9	1.7%
Cardiac Output (ml/min)	5321	4588	4623	0.8%
Heart Rate (beats/min)	72	72	72	0%
Mean Right Ventricular Pressure (mmHg)	10.6	11.1	11.1	0%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	19.5	19.3	1.0%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	24.7	24.6	0.4%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	14.6	14.5	0.7%
Pulmonary Wedge Pressure (mmHg)	9.2	13.4	13.3	0.7%
Left Heart Ejection Fraction (%)	55	36	36	0%
Right Heart Ejection Fraction (%)	62	57	57	0%
Total Number of Iterations	7	Total Output Error		2.8%

Table A.6: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Left Ventricular Systolic Dysfunction Case

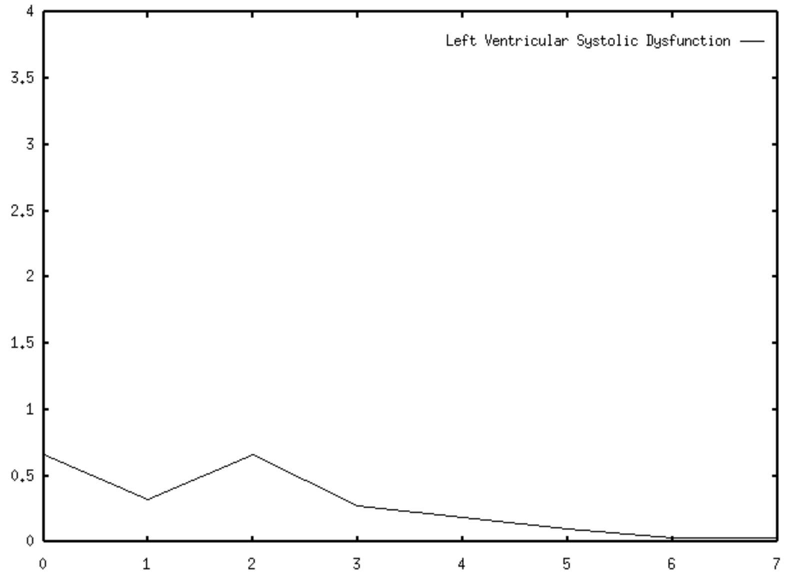

Figure A-5: Input Error Function vs. Number of Iterations in Left Ventricular Systolic Dysfunction Case

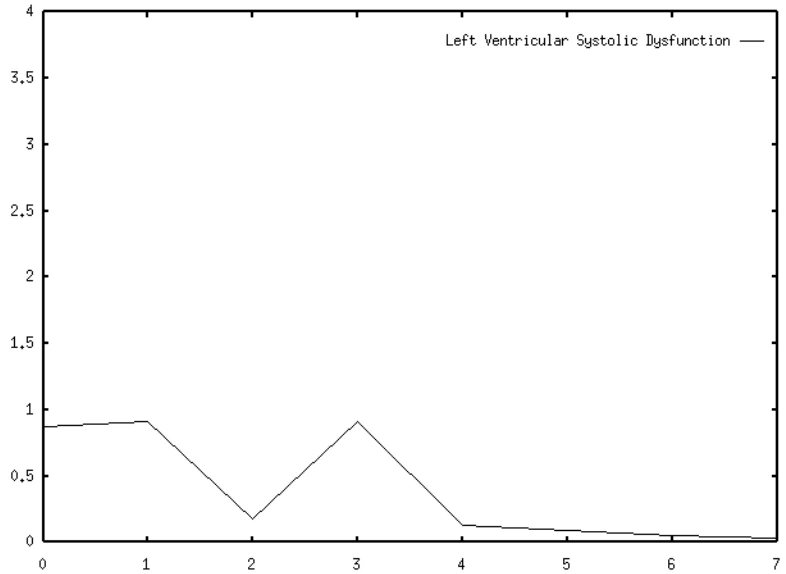

Figure A-6: Output Error Function vs. Number of Iterations in Left Ventricular Systolic Dysfunction Case

A.1.4 Arteriolar and Venous Dilation

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	72	72	0%
Total Blood Volume (ml)	5000	5000	**4742.47**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	**0.45**	13%
LVdiast	10.0	10	10	0%
Arterial	1.6	1.6	**1.57**	1.9%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	1.2	0%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**2750.0**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**925**	**917.47**	0.8%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.25**	**0.25**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.079**	1.3%
Total Number of Iterations	9	Total Input Error		13%

Table A.7: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Arteriolar and Venous Dilation Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	27.2	26.8	1.5%
Maximum Systemic Arterial Pressure (mmHg)	113.2	44.2	43.8	0.9%
Minimum Systemic Arterial Pressure (mmHg)	75.6	13.7	13.2	3.6%
Mean Central Venous Pressure (mmHg)	6.6	5.9	5.8	1.7%
Cardiac Output (ml/min)	5321	5163	5108	1.1%
Heart Rate (beats/min)	72	72	72	0%
Mean Right Ventricular Pressure (mmHg)	10.6	9.0	9.0	0%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	12.9	13.0	0.8%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	18.6	18.5	0.5%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	7.6	7.7	1.3%
Pulmonary Wedge Pressure (mmHg)	9.2	6.1	6.2	1.6%
Left Heart Ejection Fraction (%)	55	68	67	1.5%
Right Heart Ejection Fraction (%)	62	64	63	1.6%
Total Number of Iterations	9	Total Output Error		5.5%

Table A.8: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Arteriolar and Venous Dilation Case

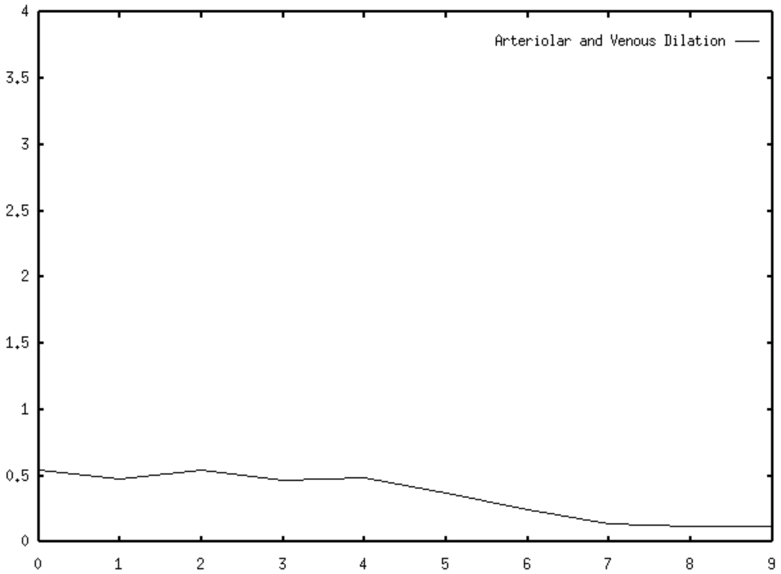

Figure A-7: Input Error Function vs. Number of Iterations in Arteriolar and Venous Dilation Case

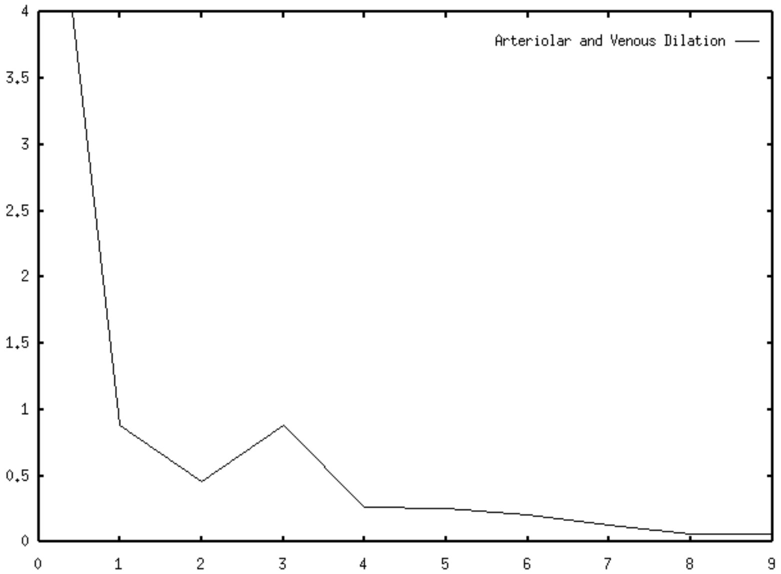

Figure A-8: Output Error Function vs. Number of Iterations in Arteriolar and Venous Dilation Case

A.2 Cases With Compensatory Effects

A.2.1 Hypertension

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**100**	**100**	0%
Total Blood Volume (ml)	5000	5000	**5268.53**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**0.30**	**0.31**	3.3%
LVdiast	10.0	**6.0**	**6.07**	1.2%
Arterial	1.6	**0.9**	**0.86**	4.4%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.20	**1.29**	7.5%
RVdiast	10.0	10.0	**10.22**	0.2%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**2200.0**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**1475**	**1443.53**	2.1%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**2.00**	**1.96**	2%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.078**	2.5%
Total Number of Iterations	24	Total Input Error		10.0%

Table A.9: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Hypertension Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	217.4	210.7	3.1%
Maximum Systemic Arterial Pressure (mmHg)	113.2	246.0	239.8	2.5%
Minimum Systemic Arterial Pressure (mmHg)	75.6	188.6	181.2	3.9%
Mean Central Venous Pressure (mmHg)	6.6	7.4	7.3	1.4%
Cardiac Output (ml/min)	5321	6427	6311	1.8%
Heart Rate (beats/min)	72	100	100	0%
Mean Right Ventricular Pressure (mmHg)	10.6	16.2	15.8	2.5%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	28.8	28.1	2.4%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	33.9	33.1	2.4%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	23.7	23.1	2.5%
Pulmonary Wedge Pressure (mmHg)	9.2	20.4	20.0	2.0%
Left Heart Ejection Fraction (%)	55	42	41	2.3%
Right Heart Ejection Fraction (%)	62	52	50	3.8%
Total Number of Iterations	24	Total Output Error		9.7%

Table A.10: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Hypertension Case

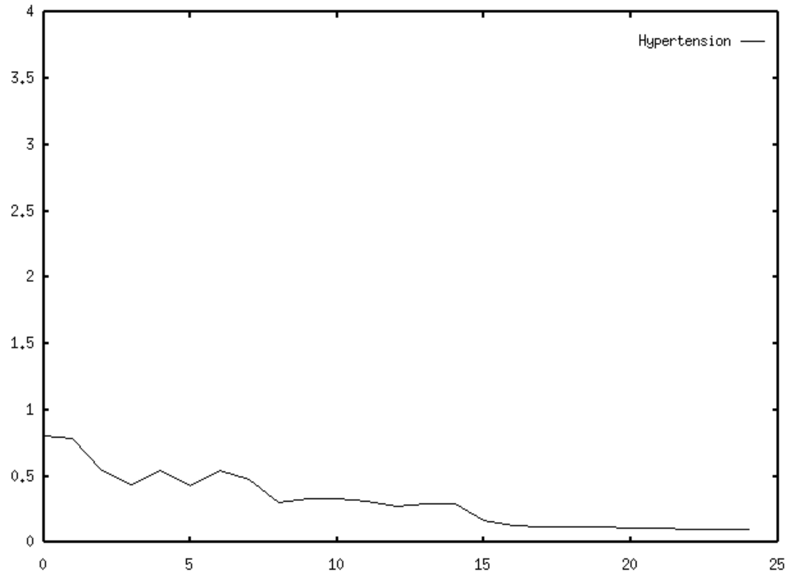

Figure A-9: Input Error Function vs. Number of Iterations in Hypertension Case

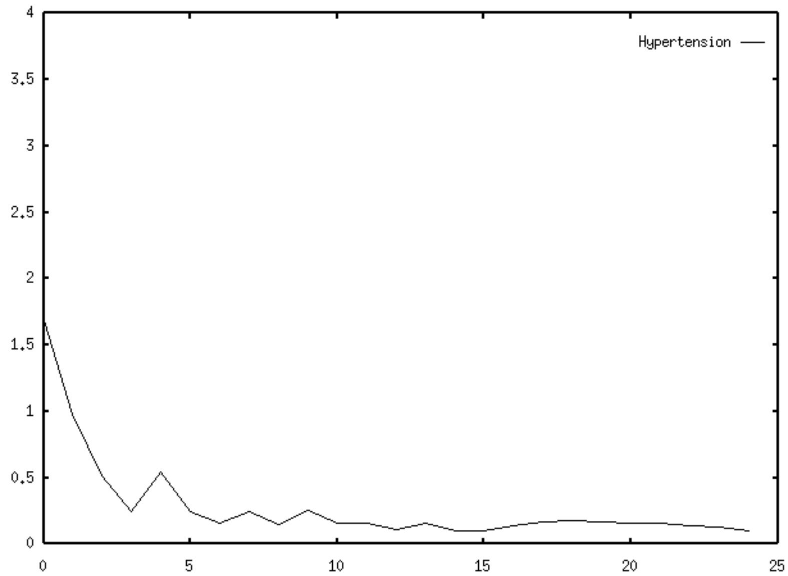

Figure A-10: Output Error Function vs. Number of Iterations in Hypertension Case

A.2.2 Right Ventricular Infarction

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**125**	**125**	0%
Total Blood Volume (ml)	5000	5000	5000	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	0.40	0%
LVdiast	10.0	10	**9.23**	7.7%
Arterial	1.6	1.6	**1.72**	7.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**5**	**6.06**	21.2%
RVdiast	10.0	**7.5**	**9.34**	24.5%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	0%
Arterial	715.0	715.0	715.0	0%
Venous	2500.0	2500	2500.0	0%
RV	15.0	15.0	15.0	0%
Pulm. Art.	90.0	90.0	90.0	0%
Pulm. Venous	490.0	490.0	490.0	0%
Total Effective Blood Volume	1175	1175	1175	0%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**1.3**	**1.46**	12.3%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.09**	12.5%
Total Number of Iterations	∞[2]	Total Input Error		38%

Table A.11: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Right Ventricular Infarction Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	81.0	85.2	5.2%
Maximum Systemic Arterial Pressure (mmHg)	113.2	88.1	91.3	3.6%
Minimum Systemic Arterial Pressure (mmHg)	75.6	74.1	78.9	6.5%
Mean Central Venous Pressure (mmHg)	6.6	8.0	7.7	3.8%
Cardiac Output (ml/min)	5321	3010	3042	1.1%
Heart Rate (beats/min)	72	125	125	0%
Mean Right Ventricular Pressure (mmHg)	10.6	7.9	8.0	1.3%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	8.1	8.3	2.5%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	9.2	9.5	3.3%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	6.9	6.9	0%
Pulmonary Wedge Pressure (mmHg)	9.2	3.6	3.5	2.8%
Left Heart Ejection Fraction (%)	55	35	33	5.7%
Right Heart Ejection Fraction (%)	62	21	19	9.5%
Total Number of Iterations	∞	Total Output Error		20.0%

Table A.12: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Right Ventricular Infarction Case

[2] ∞ means the program did not stop by itself.

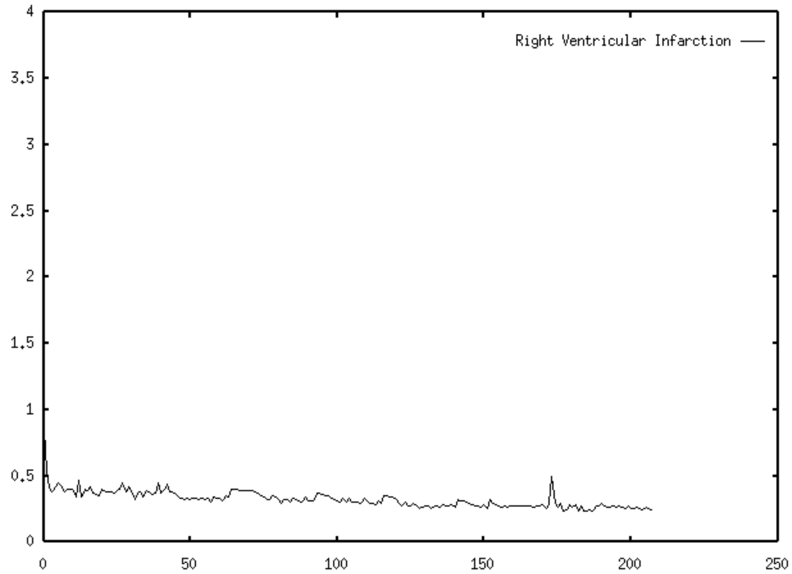

Figure A-11: Input Error Function vs. Number of Iterations
in Right Ventricular Infarction Case

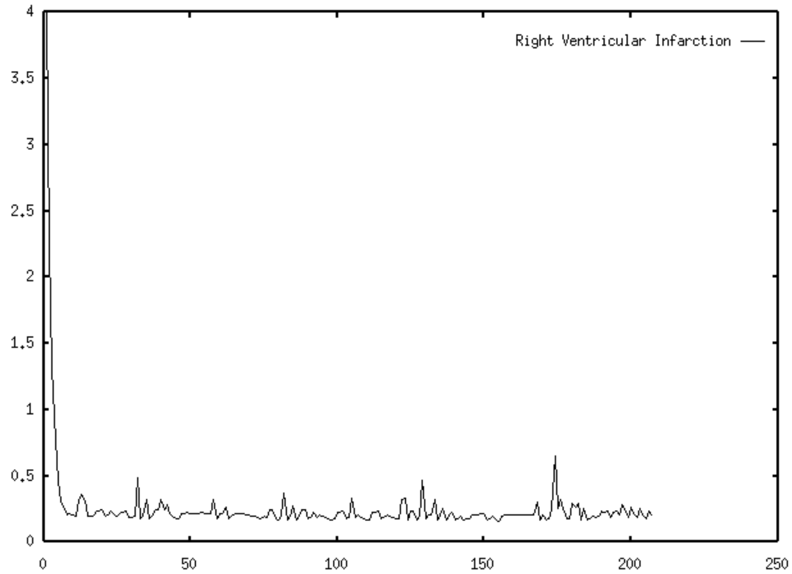

Figure A-12: Output Error Function vs. Number of Iterations
in Right Ventricular Infarction Case

A.2.3 Hypertrophic Cardiomyopathy

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**100**	**100**	0%
Total Blood Volume (ml)	5000	**6000**	**6565.28**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**0.1**	**0.11**	10%
LVdiast	10.0	**3.00**	**3.02**	0.7%
Arterial	1.6	1.6	**1.72**	7.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	**1.29**	7.5%
RVdiast	10.0	**4.0**	**4.24**	6%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**2000**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**2675**	**2740.28**	2.4%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.5**	**0.50**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.079**	1.3%
Total Number of Iterations	74	Total Input Error		14.6%

Table A.13: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Hypertrophic Cardiomyopathy Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	113.2	116.3	2.7%
Maximum Systemic Arterial Pressure (mmHg)	113.2	126.0	128.5	2.0%
Minimum Systemic Arterial Pressure (mmHg)	75.6	99.3	102.7	3.4%
Mean Central Venous Pressure (mmHg)	6.6	20.4	20.6	1.0%
Cardiac Output (ml/min)	5321	5614	5831	3.9%
Heart Rate (beats/min)	72	100	100	0%
Mean Right Ventricular Pressure (mmHg)	10.6	23.9	24.4	2.1%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	27.3	28.6	4.8%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	31.3	32.6	4.2%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	22.9	23.9	4.4%
Pulmonary Wedge Pressure (mmHg)	9.2	20.0	20.9	4.5%
Left Heart Ejection Fraction (%)	55	67	67	0%
Right Heart Ejection Fraction (%)	62	50	49	2%
Total Number of Iterations	74	Total Output Error		13.6%

Table A.14: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Hypertrophic Cardiomyopathy Case

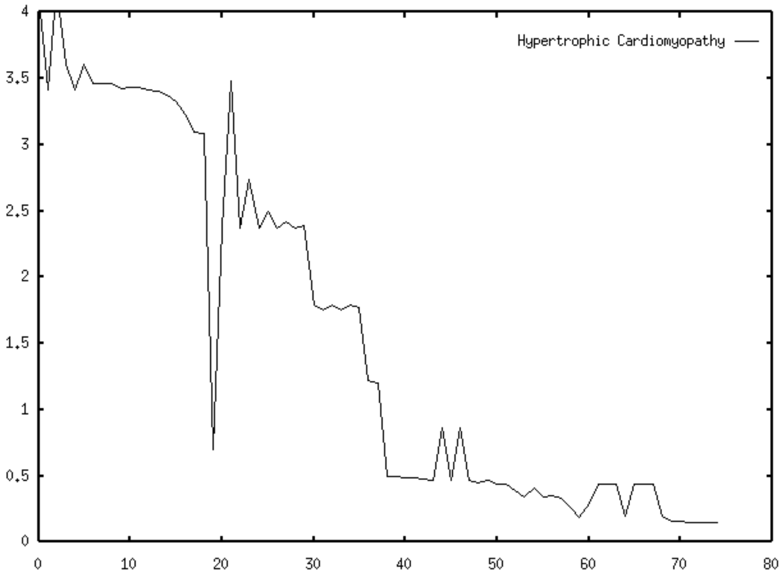

Figure A-13: Input Error Function vs. Number of Iterations in Hypertrophic Cardiomyopathy Case

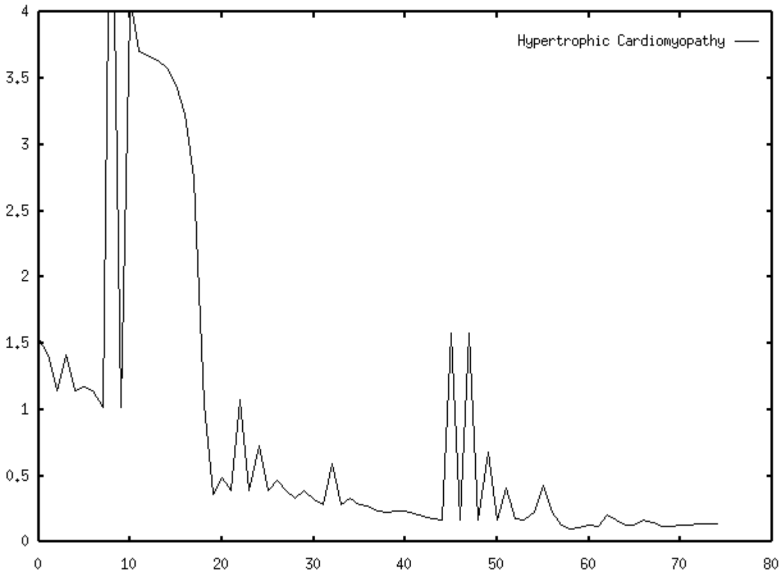

Figure A-14: Output Error Function vs. Number of Iterations in Hypertrophic Cardiomyopathy Case

A.2.4 Pulmonary Hypertension

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**120**	**120**	0%
Total Blood Volume (ml)	5000	**6000**	**6386.48**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	**0.42**	5%
LVdiast	10.0	10	**9.88**	1.2%
Arterial	1.6	1.6	**1.72**	7.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**0.4**	**0.42**	5%
RVdiast	10.0	10.0	**9.67**	3.3%
Pulm. Art.	4.30	**2.5**	4.30	72%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**2175**	**2561.48**	17.8%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.0	**0.98**	2%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**1.0**	**0.98**	2%
Total Number of Iterations	∞	Total Input Error		75.0%

Table A.15: Comparison in Pseudo-Patient Input Parameters with Estimated Input Parameters in Pulmonary Hypertension Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	155.9	154.3	1.0%
Maximum Systemic Arterial Pressure (mmHg)	113.2	158.3	170.1	7.5%
Minimum Systemic Arterial Pressure (mmHg)	75.6	137.3	138.4	0.8%
Mean Central Venous Pressure (mmHg)	6.6	11.5	12.0	4.3%
Cardiac Output (ml/min)	5321	8881	8974	1.0%
Heart Rate (beats/min)	72	120	120	0%
Mean Right Ventricular Pressure (mmHg)	10.6	60.3	60.1	0.3%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	158.3	156.7	1.0%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	169.8	163.5	3.7%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	146.0	149.4	2.3%
Pulmonary Wedge Pressure (mmHg)	9.2	13.1	13.4	2.3%
Left Heart Ejection Fraction (%)	55	46	46	0%
Right Heart Ejection Fraction (%)	62	47	47	0%
Total Number of Iterations	∞	Total Output Error		10.1%

Table A.16: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Pulmonary Hypertension Case

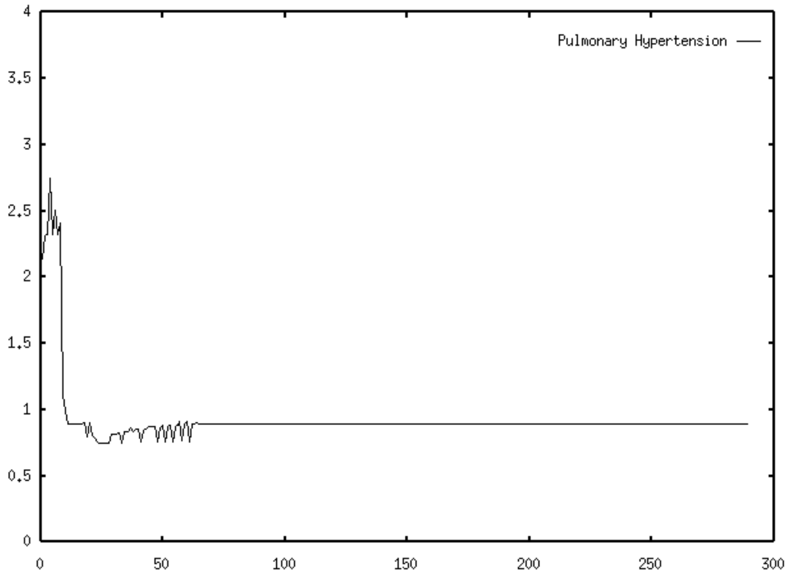

Figure A-15: Input Error Function vs. Number of Iterations in Pulmonary Hypertension Case

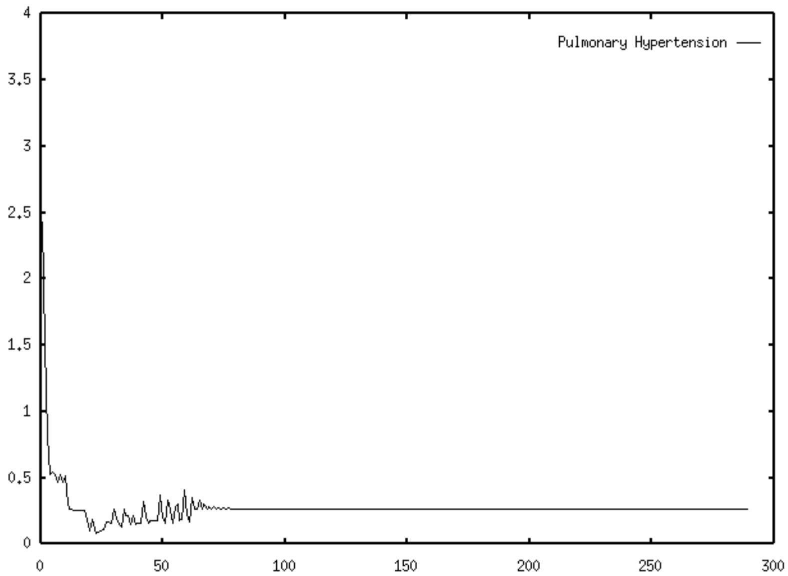

Figure A-16: Output Error Function vs. Number of Iterations in Pulmonary Hypertension Case

A.2.5 Large AV Fistula

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**120**	**120**	0%
Total Blood Volume (ml)	5000	**5500**	**7319.73**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**0.20**	**0.23**	15%
LVdiast	10.0	10	**9.74**	2.6%
Arterial	1.6	1.6	**1.63**	1.9%
Venous	100.0	**70.0**	100.0	42.9%
RVsyst	1.20	**0.6**	**0.67**	11.7%
RVdiast	10.0	10	**9.66**	3.4%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**1500**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**2675**	**3494.73**	30.6%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.10**	**0.10**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**0.04**	**0.04**	0%
Total Number of Iterations	23	Total Input Error		55.7%

Table A.17: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in AV Fistula Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	70.6	69.8	1.1%
Maximum Systemic Arterial Pressure (mmHg)	113.2	116.5	113.7	2.4%
Minimum Systemic Arterial Pressure (mmHg)	75.6	35.7	36.1	1.1%
Mean Central Venous Pressure (mmHg)	6.6	25.5	26.0	2.0%
Cardiac Output (ml/min)	5321	27362	26804	2.0%
Heart Rate (beats/min)	72	120	120	0%
Mean Right Ventricular Pressure (mmHg)	10.6	34.9	35.4	1.4%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	46.2	45.9	0.6%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	61.2	60.4	1.3%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	30.1	30.1	0%
Pulmonary Wedge Pressure (mmHg)	9.2	28.3	28.5	0.7%
Left Heart Ejection Fraction (%)	55	86	85	1.2%
Right Heart Ejection Fraction (%)	62	82	80	2.4%
Total Number of Iterations	23	Total Output Error		6.1%

Table A.18: Comparison of Output Variables of Pseudo-Patient with those of Simulator in AV Fistula Case

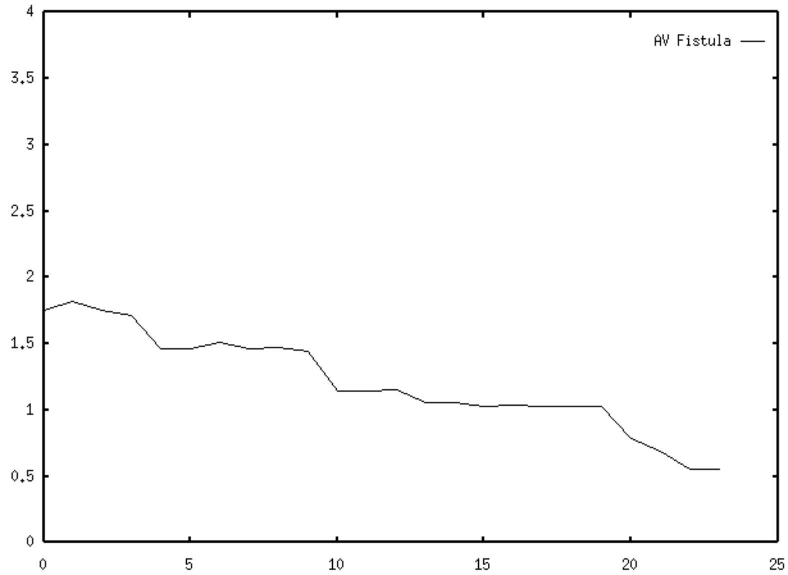

Figure A-17: Input Error Function vs. Number of Iterations in AV Fistula Case

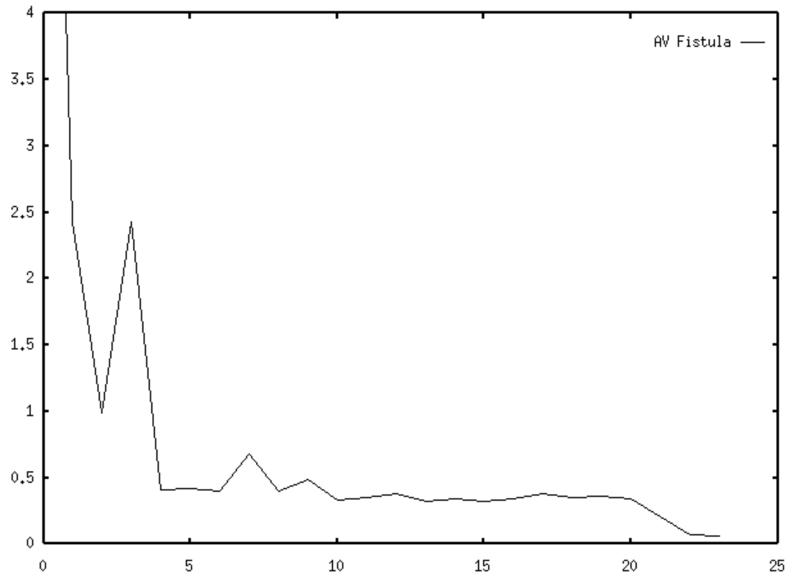

Figure A-18: Output Error Function vs. Number of Iterations in AV Fistula Case

A.2.6 Dilated Cardiomyopathy and Bi-Ventricular Congestive Heart Failure

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**125**	**125**	0%
Total Blood Volume (ml)	5000	**6300**	**6684**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**1.2**	**1.6**	33.3%
LVdiast	10.0	**5.0**	**6.2**	24%
Arterial	1.6	**1.0**	**1.1**	10%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**3.6**	**4.4**	22.2%
RVdiast	10.0	**11.0**	**13.3**	20.9%
Pulm. Art.	4.30	**2.5**	**2.7**	8%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	**50**	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**2200**	2500.0	
RV	15.0	**50.0**	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**2705**	**2859.04**	5.7%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.9**	**0.88**	2.2%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**0.14**	**0.14**	0%
Total Number of Iterations	82	Total Input Error		52.0%

Table A.19: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Dilated Cardiomyopathy and Bi-Ventricular Congestive Heart Failure Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	95.1	96.6	1.6%
Maximum Systemic Arterial Pressure (mmHg)	113.2	108.4	110.0	1.5%
Minimum Systemic Arterial Pressure (mmHg)	75.6	79.6	81.4	2.3%
Mean Central Venous Pressure (mmHg)	6.6	18.1	18.4	1.7%
Cardiac Output (ml/min)	5321	5247.7	5475	4.3%
Heart Rate (beats/min)	72	125	125	0%
Mean Right Ventricular Pressure (mmHg)	10.6	31.0	31.5	1.6%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	44.3	45.4	2.5%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	49.5	50.5	2.0%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	38.6	40.0	3.6%
Pulmonary Wedge Pressure (mmHg)	9.2	32.5	33.5	3.1%
Left Heart Ejection Fraction (%)	55	19	19	0%
Right Heart Ejection Fraction (%)	62	15	14	0%
Total Number of Iterations	82	Total Output Error		16.8%

Table A.20: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Dilated Cardiomyopathy and Bi-Ventricular Congestive Heart Failure Case

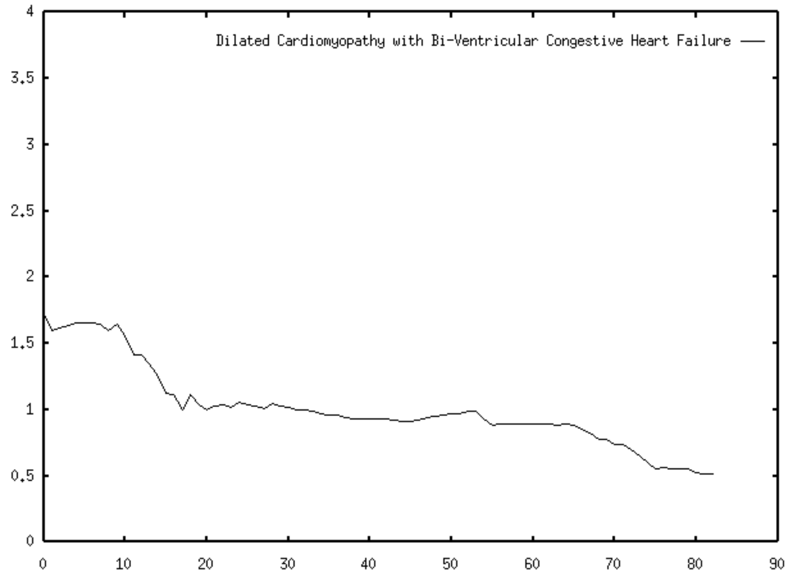

Figure A-19: Input Error Function vs. Number of Iterations in Dilated Cardiomyopathy and Bi-ventricular Congestive Heart Failure Case

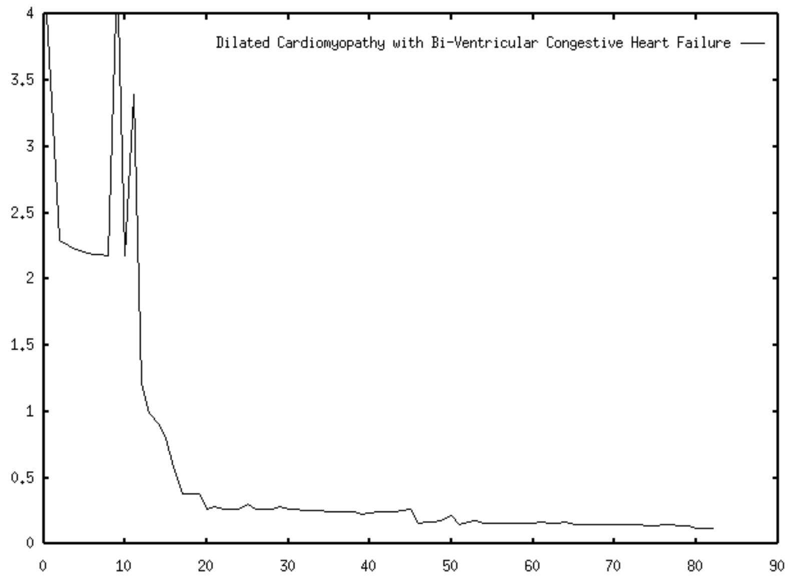

Figure A-20: Output Error Function vs. Number of Iterations in Dilated Cardiomyopathy and Bi-ventricular Congestive Heart Failure Case

A.2.7 Acute Myocardial Infarction with Bi-Ventricular Heart Failure

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**110**	**110**	0%
Total Blood Volume (ml)	5000	5000	**5748.93**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**2.5**	**2.6**	4%
LVdiast	10.0	**8.0**	**8.21**	2.6%
Arterial	1.6	**0.5**	**0.50**	0%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**4.0**	**4.4**	10%
RVdiast	10.0	**15**	**15.7**	4.7%
Pulm. Art.	4.30	**2.0**	**1.9**	5%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**1800.0**	2500.0	
RV	15.0	**25.0**	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**1865**	**1653**	11.4%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**1.40**	**1.34**	4.3%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	**0.077**	3.8%
Total Number of Iterations	77	Total Input Error		14.6%

Table A.21: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Acute Myocardial Infarction and Bi-ventricular Heart Failure Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	78.3	76.3	2.6%
Maximum Systemic Arterial Pressure (mmHg)	113.2	94.8	93.3	1.6%
Minimum Systemic Arterial Pressure (mmHg)	75.6	59.3	56.8	4.2%
Mean Central Venous Pressure (mmHg)	6.6	10.0	10.5	5%
Cardiac Output (ml/min)	5321	3054	3066	0.4%
Heart Rate (beats/min)	72	110	110	0%
Mean Right Ventricular Pressure (mmHg)	10.6	21.2	21.4	0.9%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	35.0	34.4	1.7%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	40.0	39.1	2.3%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	31.4	30.8	1.9%
Pulmonary Wedge Pressure (mmHg)	9.2	31.1	30.6	1.6%
Left Heart Ejection Fraction (%)	55	10	10	0%
Right Heart Ejection Fraction (%)	62	12	12	0%
Total Number of Iterations	77	Total Output Error		10.1%

Table A.22: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Acute Myocardial Infarction and Bi-Ventricular Heart Failure Case

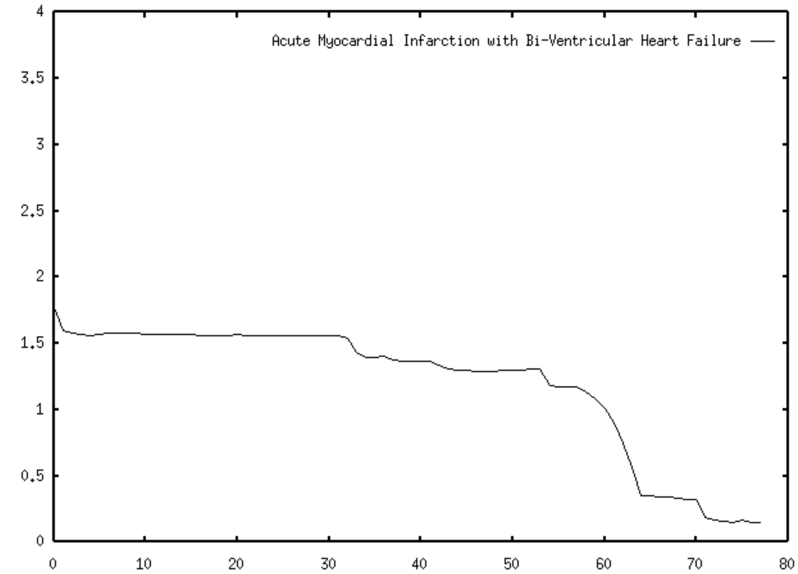

Figure A-21: Input Error Function vs. Number of Iterations in
Acute Myocardial Infarction and Bi-ventricular Heart Failure Case

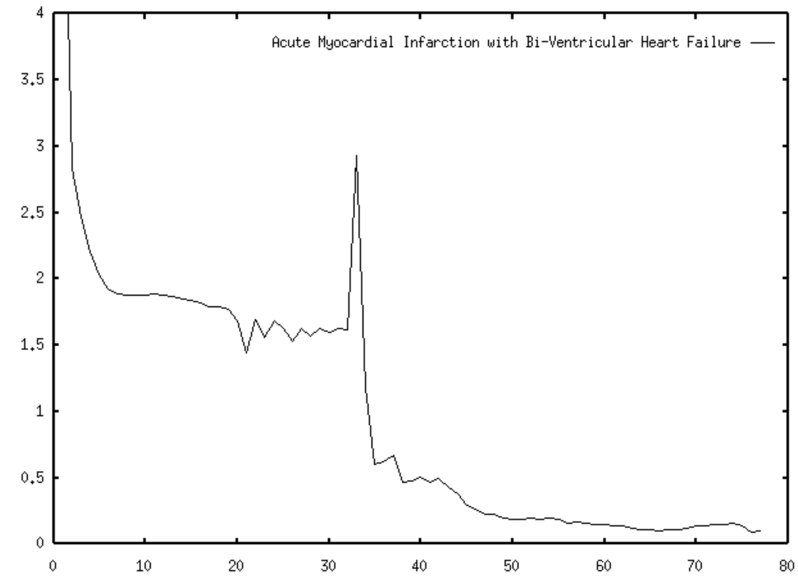

Figure A-22: Output Error Function vs. Number of Iterations in
Acute Myocardial Infarction and Bi-Ventricular Heart Failure Case

A.2.8 Septic Shock

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**150**	**150**	0%
Total Blood Volume (ml)	5000	5000	**4519.74**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	**0.5**	**0.6**	10%
LVdiast	10.0	10	**10.4**	4%
Arterial	1.6	1.6	**1.64**	2.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	1.2	0%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**3000.0**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**675**	**694.74**	2.9%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**0.25**	**0.25**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	0.08	0%
Total Number of Iterations	51	Total Input Error		14.2%

Table A.23: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Septic Shock Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	30.4	30.8	1.3%
Maximum Systemic Arterial Pressure (mmHg)	113.2	39.4	39.8	1.0%
Minimum Systemic Arterial Pressure (mmHg)	75.6	21.3	21.6	1.4%
Mean Central Venous Pressure (mmHg)	6.6	3.9	4.0	2.6%
Cardiac Output (ml/min)	5321	6483	6662	2.8%
Heart Rate (beats/min)	72	150	150	0%
Mean Right Ventricular Pressure (mmHg)	10.6	8.2	8.5	3.7%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	13.1	13.4	2.3%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	15.9	16.3	2.5%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	9.5	9.8	3.2%
Pulmonary Wedge Pressure (mmHg)	9.2	4.5	4.7	4.4%
Left Heart Ejection Fraction (%)	55	54	53	1.9%
Right Heart Ejection Fraction (%)	62	53	53	0%
Total Number of Iterations	51	Total Output Error		10.7%

Table A.24: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Septic Shock Case

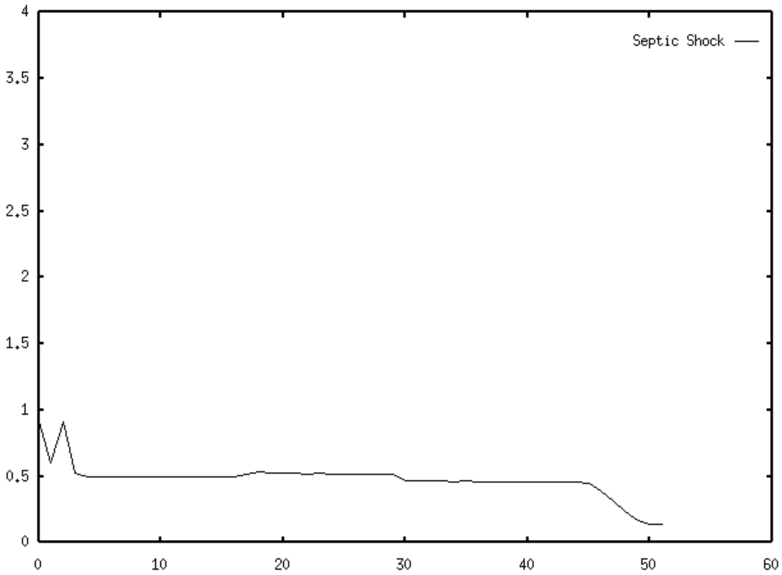

Figure A-23: Input Error Function vs. Number of Iterations in Septic Shock Case

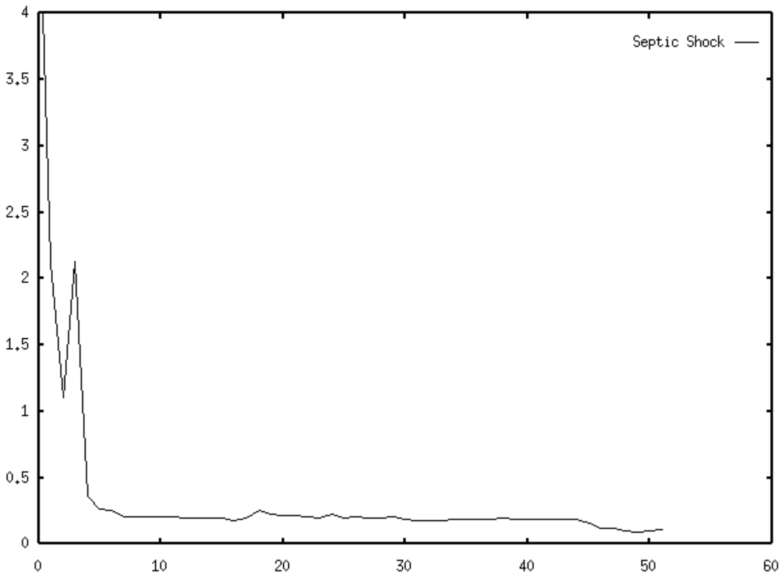

Figure A-24: Output Error Function vs. Number of Iterations in Septic Shock Case

A.2.9 Pulmonary Hypertension with Right Heart Failure

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**152**	**152**	0%
Total Blood Volume (ml)	5000	**6500**	**6712.44**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	**0.41**	2.5%
LVdiast	10.0	10	**9.7**	3%
Arterial	1.6	**0.80**	**0.80**	0%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**4**	**4.06**	1.5%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	**2200**	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**2975**	**2887.44**	2.9%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	**1.4**	**1.38**	1.4%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**1**	**0.96**	4%
Total Number of Iterations	50	Total Input Error		6.6%

Table A.25: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Pulmonary Hypertension and Right Heart Failure Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	95.5	92.3	3.4%
Maximum Systemic Arterial Pressure (mmHg)	113.2	106.3	102.8	3.3%
Minimum Systemic Arterial Pressure (mmHg)	75.6	85.1	82.1	3.5%
Mean Central Venous Pressure (mmHg)	6.6	22.8	22.1	3.1%
Cardiac Output (ml/min)	5321	3163	3115	1.5%
Heart Rate (beats/min)	72	152	152	0%
Mean Right Ventricular Pressure (mmHg)	10.6	38.7	37.2	3.9%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	54.2	51.7	4.6%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	55.9	53.3	4.7%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	52.4	49.9	4.8%
Pulmonary Wedge Pressure (mmHg)	9.2	3.7	3.8	2.7%
Left Heart Ejection Fraction (%)	55	26	26	0%
Right Heart Ejection Fraction (%)	62	7.5	7.6	1.3%
Total Number of Iterations	50	Total Output Error		12.8%

Table A.26: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Pulmonary Hypertension and Right Heart Failure Case

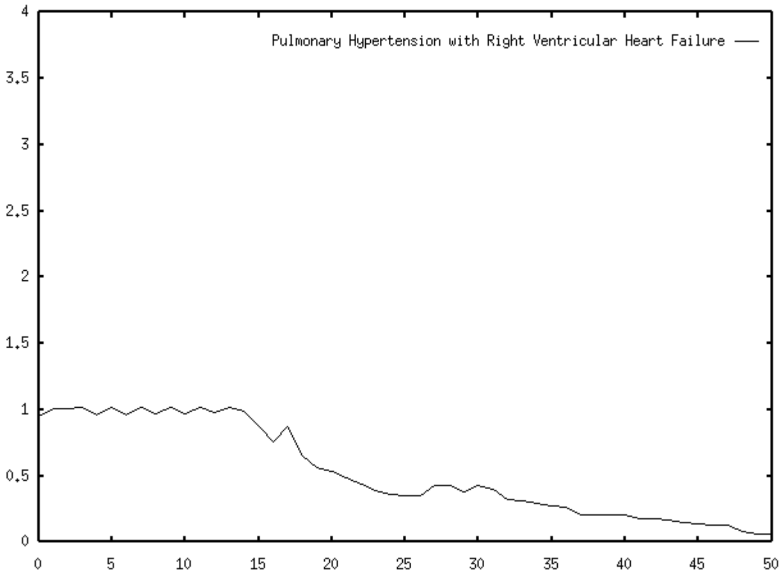

Figure A-25: Input Error Function vs. Number of Iterations in Pulmonary Hypertension and Right Heart Failure Case

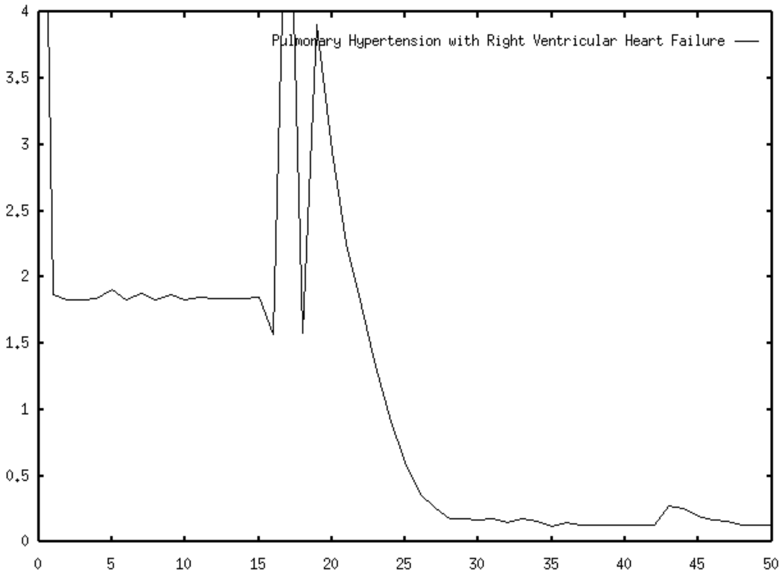

Figure A-26: Output Error Function vs. Number of Iterations in Pulmonary Hypertension and Right Heart Failure Case

A.2.10 Mitral Stenosis I

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**100**	**100**	0%
Total Blood Volume (ml)	5000	**5500**	**5530.11**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	**0.42**	5%
LVdiast	10.0	10	**2.69**	73.1%
Arterial	1.6	1.6	**1.49**	6.9%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**0.9**	**0.91**	1.1%
RVdiast	10.0	10.0	**9.67**	3.3%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**1675**	**1705.11**	1.8%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	**0.2**	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.0	**0.98**	2%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**0.30**	**0.29**	3.3%
Total Number of Iterations	30	Total Input Error		120.3%

Table A.27: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Mitral Stenosis Case I

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	93.9	92.3	1.7%
Maximum Systemic Arterial Pressure (mmHg)	113.2	107.2	104.0	3.0%
Minimum Systemic Arterial Pressure (mmHg)	75.6	80.7	78.6	2.6%
Mean Central Venous Pressure (mmHg)	6.6	7.8	8.2	5.1%
Cardiac Output (ml/min)	5321	5286	5258	0.5%
Heart Rate (beats/min)	72	100	100	0%
Mean Right Ventricular Pressure (mmHg)	10.6	26.0	26.1	0.4%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	59.1	58.6	0.8%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	63.7	63.3	0.8%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	54.1	53.7	0.7%
Pulmonary Wedge Pressure (mmHg)	9.2	33.1	33.1	0%
Left Heart Ejection Fraction (%)	55	47	47	0%
Right Heart Ejection Fraction (%)	62	41	41	0%
Total Number of Iterations	30	Total Output Error		7.3%

Table A.28: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Mitral Stenosis Case I

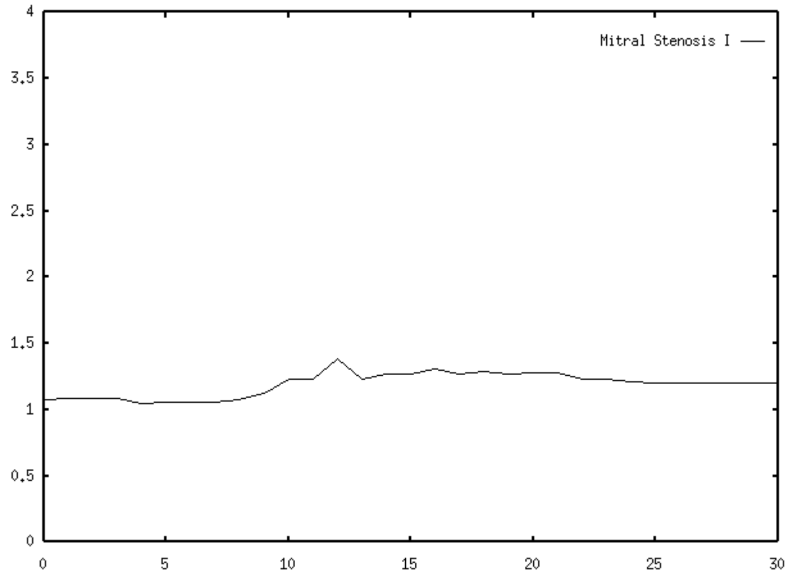

Figure A-27: Input Error Function vs. Number of Iterations in Mitral Stenosis Case I

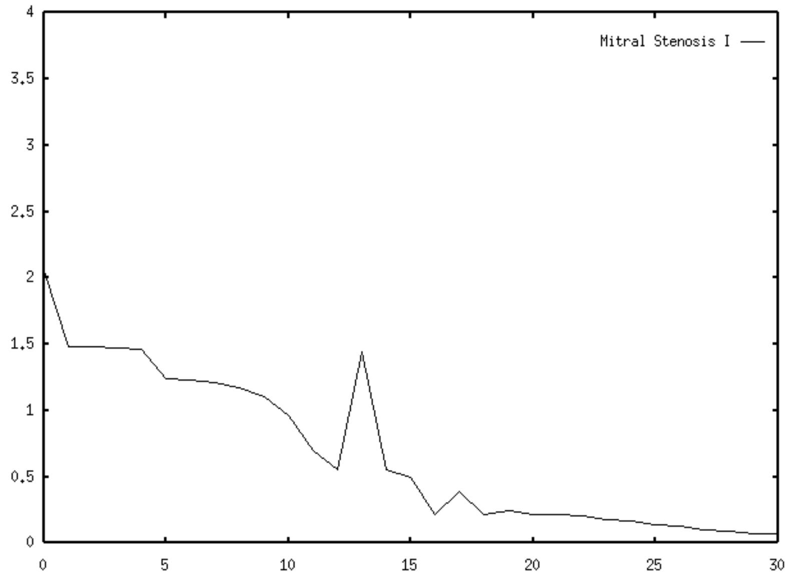

Figure A-28: Output Error Function vs. Number of Iterations in Mitral Stenosis Case I

A.2.11 Mitral Stenosis II

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**100**	**100**	0%
Total Blood Volume (ml)	5000	**5500**	**5544.33**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	**0.43**	7.5%
LVdiast	10.0	10	10	0%
Arterial	1.6	1.6	**1.8**	12.5%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	**0.9**	**0.93**	3.3%
RVdiast	10.0	10.0	**10.6**	6%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**1675**	**1719.33**	2.6%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	**0.2**	**0.19**	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.0	**0.98**	0%
Venous	0.01	0.01	0.01	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	**0.30**	**0.29**	3.3%
Total Number of Iterations	27	Total Input Error		16.5%

Table A.29: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Mitral Stenosis Case II

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	93.9	96.8	3.1%
Maximum Systemic Arterial Pressure (mmHg)	113.2	107.2	109.3	2.0%
Minimum Systemic Arterial Pressure (mmHg)	75.6	80.7	84.1	4.2%
Mean Central Venous Pressure (mmHg)	6.6	7.8	7.8	0%
Cardiac Output (ml/min)	5321	5286	5529	4.6%
Heart Rate (beats/min)	72	100	100	0%
Mean Right Ventricular Pressure (mmHg)	10.6	26.0	26.4	1.5%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	59.1	60.4	2.2%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	63.7	65.2	2.4%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	54.1	55.2	2.0%
Pulmonary Wedge Pressure (mmHg)	9.2	33.1	33.6	1.5%
Left Heart Ejection Fraction (%)	55	47	46	2.1%
Right Heart Ejection Fraction (%)	62	41	41	0%
Total Number of Iterations	27	Total Output Error		11.1%

Table A.30: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Mitral Stenosis Case II

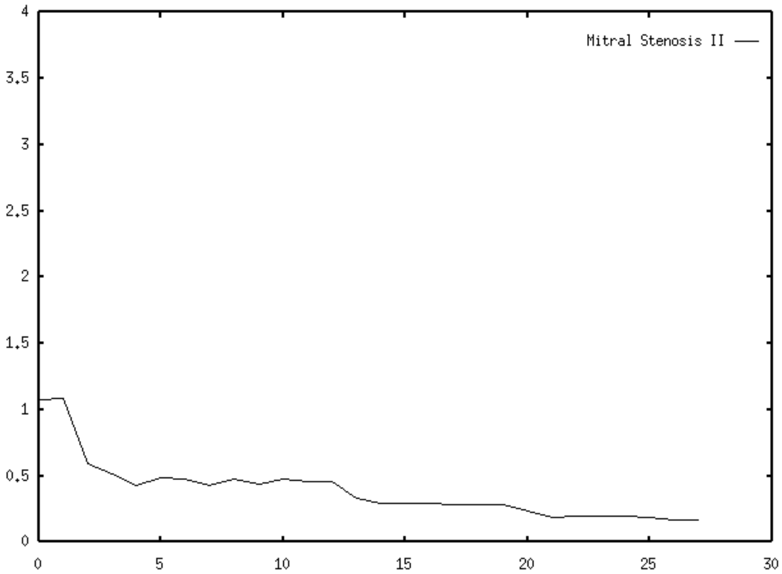

Figure A-29: Input Error Function vs. Number of Iterations in Mitral Stenosis Case II

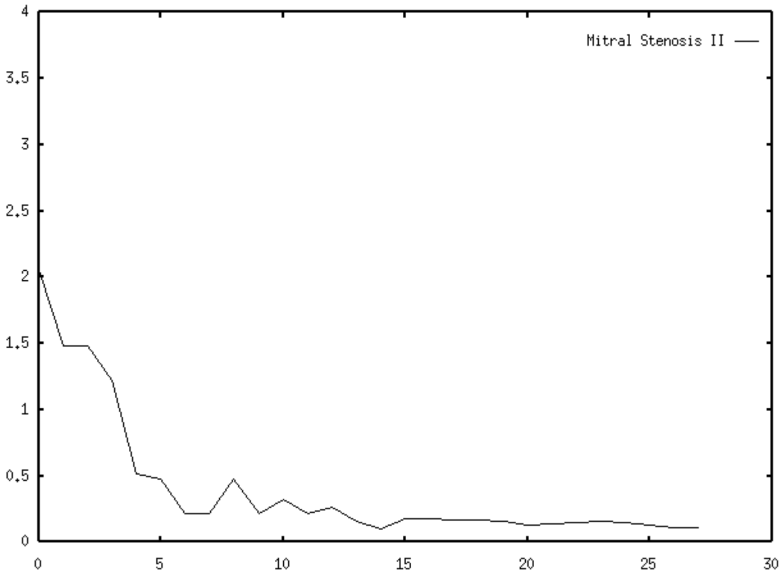

Figure A-30: Output Error Function hvjkcxvs. Number of Iterations in Mitral Stenosis Case II

A.2.12 Tricuspid Stenosis

Parameter	Normal	Pseudo-Patient	Estimation	Error
Heart Rate (beats/min)	72	**120**	**120**	0%
Total Blood Volume (ml)	5000	**5500**	**5525.11**	
Trans-thoracic Pressure (mmHg)	-4	-4	-4	0%
Capacitances: (ml/mmHg)				
LVsyst	0.4	0.4	0.4	0%
LVdiast	10.0	10	10.0	0%
Arterial	1.6	1.6	1.6	0%
Venous	100.0	100.0	100.0	0%
RVsyst	1.20	1.2	1.2	0%
RVdiast	10.0	10.0	10.0	0%
Pulm. Art.	4.30	4.30	4.30	0%
Pulm. Venous	8.40	8.40	8.40	0%
Zero-Pressure Volumes: (ml)				
LV	15.0	15.0	15.0	
Arterial	715.0	715.0	715.0	
Venous	2500.0	2500	2500.0	
RV	15.0	15.0	15.0	
Pulm. Art.	90.0	90.0	90.0	
Pulm. Venous	490.0	490.0	490.0	
Total Effective Blood Volume	1175	**1675**	**1700.11**	1.5%
Resistances: (mmHg*sec/ml)				
LV Inflow	0.01	0.01	0.01	0%
LV Outflow	0.006	0.006	0.006	0%
Microvascular	1.0	1.0	**0.98**	2%
Venous	0.01	**0.1**	**0.1**	0%
RV Outflow	0.003	0.003	0.003	0%
Pulmonary	0.08	0.08	0.08	0%
Total Number of Iterations	23	Total Input Error		4.3%

Table A.31: Comparison of Pseudo-Patient Input Parameters with Estimated Input Parameters in Tricuspid Stenosis Case

Measurable Variables	Normal	Pseudo-Patient	Simulator Output	Error
Mean Systemic Arterial Pressure (mmHg)	94.3	86.2	86.7	0.6%
Maximum Systemic Arterial Pressure (mmHg)	113.2	95.5	96.0	0.5%
Minimum Systemic Arterial Pressure (mmHg)	75.6	76.7	77.4	0.9%
Mean Central Venous Pressure (mmHg)	6.6	13.0	13.2	1.5%
Cardiac Output (ml/min)	5321	4409	4436	0.6%
Heart Rate (beats/min)	72	120	120	0%
Mean Right Ventricular Pressure (mmHg)	10.6	5.9	6.1	3.4%
Mean Pulmonary Arterial Pressure (mmHg)	16.3	10.9	11.0	0.9%
Maximum Pulmonary Arterial Pressure (mmHg)	22.1	13.5	13.6	0.7%
Minimum Pulmonary Arterial Pressure (mmHg)	10.7	7.9	8.0	1.3%
Pulmonary Wedge Pressure (mmHg)	9.2	5.0	5.1	2%
Left Heart Ejection Fraction (%)	55	41	41	0%
Right Heart Ejection Fraction (%)	62	50	50	0%
Total Number of Iterations	23	Total Output Error		4.6%

Table A.32: Comparison of Output Variables of Pseudo-Patient with those of Simulator in Tricuspid Stenosis Case

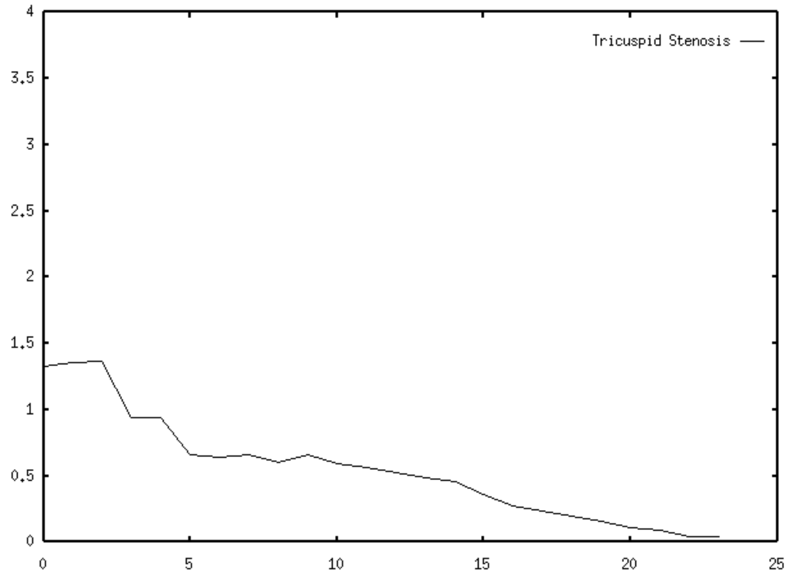

Figure A-31: Input Error Function vs. Number of Iterations in Tricuspid Stenosis Case

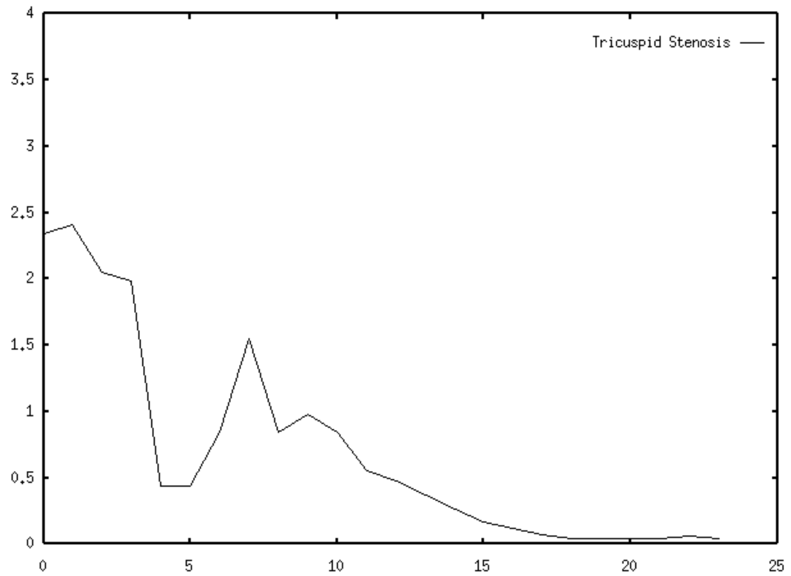

Figure A-32: Output Error Function vs. Number of Iterations in Tricuspid Stenosis Case

Bibliography

[1] N. Abramson. Adverse occurrences in intensive care units. *JAMA*, 244:1582–1584, 1980.

[2] R. Beeuwkes and N. Braslow. The cardiovascular analog trainer: A real time physiological simulator. *Physiology Teacher*, 3(3):4–7, 1974.

[3] B. Berwick, J. Jaroslav, T. Knight, G. J. Sussman, P. H. Winston, S. Ullman, and K. Yip. What we think. Web Site: *http://www.ai.mit.edu/people/phw/state.html*.

[4] K. Campbell, M. Zeglen, T. Kagehiro, and R. Harriet. A pulsatile cardiovascular computer model for teaching heart-blood vessel interaction. *Physiologist*, 25(3):155–162, 1982.

[5] E. Charniak and D. McDemott. *Introduction to Artificial Intelligence*. Addison-Wesley, Reading, MA, 1985.

[6] R. Davis and D. B. Lenat. *Knowledge-Based Systems in Artificial Intelligence*. McGraw-Hill International Book Company, 1982.

[7] T. L. Davis. Teaching physiology through interactive simulation of hemodynamics. Master's project, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, February 1991.

[8] J. G. Defares, J. J. Osborne, and H. H. Hiroshi. Theoretical synthesis of the cardiovascular system. *Acta Physiol. Pharm. Neerl.*, 12:189–265, 1963.

[9] L. M. Fagan, E. H. Shortliffe, and B. G. Buchanan. Computer-based medical decision making: from MYCIN to VM. Stanford heuristic programming project, Stanford University, 1979. Working Paper: HPP-79-18.

[10] E. Feigenbaum, P. McCorduck, and H. P. Nii. *The rise of the Expert System*. Times Books, 1988.

[11] M. B. First, B. J. Weimer, S. McLinden, and R. A. Miller. Localize: Computer-assisted localization of peripheral nervous system lesions. *Comput. Biomed. Res.*, pages 525+, 1982.

[12] J. S. Gravenstein and R. S. Newbower. An integrated approach to monitoring. Woburn, MA, 1983. Butterworth's.

[13] W. E. L. Grimson and R. S. Patil. *AI in the 1980s and Beyond*. The MIT Press, Cambridge, Massachusetts 02142, 1987.

[14] F. S. Grodins. Integrative cardiovascular physiology: A mathematical synthesis of cardiac and blood vessel hemodynamics. *The Quarterly Review of Biology*, 34(2):93–116, June 1959.

[15] A. C. Guyton. *Circulatory Physiology: Cardiac Output and its Regulation*. W. B. Saunders Company, 2nd edition, 1973.

[16] R. D. Hawkins. Artificial intelligence from the system engineer's viewpoint. In W.M.Holmes, editor, *Artificial Intelligence and Simulation*, pages 460–463, San Diego, CA, 1985. Society for Computer Simulation.

[17] D. E. Hirsch. An expert system for diagnosing gait for cerebral palsy patients. Technical Report MIT/LCS/TR-388, Massachusetts Institute of Technology, Laboratory for Computer Science, Cambridge, MA, May 1987.

[18] S. Katz, R. G. Hollingsworth, and J. G. Blackburn. Computer simulation in the physiology student laboratory. *Physiologist*, 21(6):41–44, 1978.

[19] L. C. Kingsland. The evaluation of medical expert systems: Experience with the ai/rheum knowledge-based consultant system in rheumatology. In *Ninth Symposium on Computer Applications in Medical Care*, pages 292–295, Washington, DC, 1985.

[20] J. C. Kunz. Analysis of physiological behavior using a casual model based on first principles. In *Proc. 3rd Natl. Conf. in Artificial Intelligence*, pages 225+, San Mateo, CA, 1983. Morgan Kaufmann.

[21] W. J. Long, M. G. Criscitiello, and R. Jayes. Development and use of a causal model for reasoning about heart failure. In P. L. Miller, editor, *Selected Topics in medical Artificial Intelligence*, pages 40+, New York, 1988. Springer-Verlag.

[22] W. J. Long, S. Naimi, and M. G. Criscitiello. Evaluation of a new method for cardiovascular reasoning. *Journal of the American Medical Informatics Association*, (1):127–141, 1994.

[23] F. M. Melchior, R. S. Srinivasan, and J. B. Charles. Mathematical model of human cardiovascular system for simulation of orthostatic response. *Am. J. Physiol.*, 262:H1920–H1933, 1992.

[24] C. Meredith and J. Edworthy. Are there too many alarms in the intensive care unit? *Journal of Advanced Nurses*, 21:15–20, 1995.

[25] R. A. Miller, H. E. Pople, and J. D. Myers. INTERNIST-I, an experimental computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine*, 307:468–476, 1982.

[26] A. P. Morgan and A. J. Sommese. A homotopy for solving general polynomial systems that respects m-homogeneous structures. *Applied Mathematics Computation*, 24:101–113, 1987.

[27] C. Oleson. INTERNIST: A computer-based consultation, computer networking in the university: Success and potential. In *Proc. of the EDUCOM Fall Conference*, 1976.

[28] S. G. Pauker, G. A. Gorry, J. P. Kassirer, and W. B. Schwartz. Toward the simulation of clinical cognition: Taking a present illness by computer. *The American Journal of Medicine*, 60, June 1976.

[29] H. B. Rubins. Complications of care in a medical intensive care unit. *Journal of General Internal Medicine*, 5:104–109, 1990.

[30] S. Russell and P. Norvig. *Artificial Intelligence: A modern Approach*, chapter 4.4, pages 113–114. Prentice Hall, 1995.

[31] G. W. Rutledge. *Dynamic Selection of Models.* PhD dissertation, Stanford University, March 1995.

[32] E. H. Shortliffe. *Computer-based Medical Consultations: MYCIN.* Elsevier, 1976.

[33] E. H. Shortliffe, B. G. Buchanan, and E. A. Feigenbaum. Knowledge engineering for medical decision making: A review of computer-based clinical decision aids. *Proceedings of IEEE*, September 1979.

[34] K. Sungawa and K. Sagawa. Models of ventricular contraction based on time-varying elastance. *Critical Reviews in Biomedical Engineering*, 7(3):193–228, February 1982.

[35] P. Szolovits. Knowledge-based systems. In A. R. Meyer, J. V. Guttag, R. L. Rivest, and P. Szolovits, editors, *Research Directions in Computer Science: An MIT Perspective*, pages 317–370, Cambridge, MA, 1991. MIT Press.

[36] P. Szolovits and S. G. Pauker. Categorical and probalistic reasoning in medical diagnosis. *Artificial Intelligence*, 11:115–144, 1978.

[37] S. Uckun. Model-based reasoning in biomedicine. *Critical Reviews in Biomedical Engineering*, 19(4):261–292, 1992.

[38] D. A. Waterman. *A Guide to Expert Systems.* Addison-Wesley, 1985.

[39] J. B. Wyngaarden and L. H. Smith. *Textbook of Medicine.* W. B. Saunders Company, 18th edition, 1988.

[40] R. Zhao and S. Ma. 3D reconstruction of high degree planar curves. In *Proceedings of Asian Conference on Computer Vision*, pages 474–477, Osaka, Japan, November 1993.

[41] R. Zhao, R. G. Mark, and P. Szolovits. A model-based expert system for interpretation of hemodynamic data from ICU patients. In *Computers in Cardiology*, pages 585–588, Indianapolis, Indiana, 1996.