ABSTRACT

Title of Document: ***HEALTH CARE MANAGEMENT SYSTEM FOR DIABETES MELLITUS: A MODEL-BASED SYSTEMS ENGINEERING FRAMEWORK***

.

Iakovos M. Katsipis, Master of Science, 2015

Directed By: Professor John S. Baras,
Institute for Systems Research and
Department of Electrical and Computer Engineering

The present thesis develops a framework for Health Care Management Systems using modern Model-Based Systems Engineering methodologies and applies it to Diabetes Mellitus. The desired architecture of such systems is described. Tests and interventions, including Health Care IT, used for Diabetes 2 diagnosis and treatment, are described and modeled. A Controlled Markov Chain model for the progression of Diabetes Mellitus with three states, three diagnostic tests, ten interventions, three patient types, is developed. Evaluation metrics for healthcare quality and associated costs are developed. Using these metrics and disease models, two methods for tradeoff analysis between healthcare quality and costs are developed and analyzed. One is an exhaustive Monte Carlo simulation and the other utilizes multi-criteria optimization with full state information. The latter obtains similar results as the

former at a fraction of the time. Practical examples illustrate the powerful capabilities of the framework. Future research directions and extensions are described.

HEALTH CARE MANAGEMENT SYSTEM FOR DIABETES MELLITUS:

A MODEL-SYSTEM ENGINEERING FRAMEWORK

By

Iakovos M. Katsipis.

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2015

Advisory Committee:
Professor John S. Baras, Chair / Advisor
Professor Jeffrey W. Herrmann
Professor Guangming Zhang

Dedication

To my family, to my advisor and all the people that supported me in this endeavor.

Acknowledgements

I would like to sincerely thank and express my gratitude to my advisor Professor John S. Baras for giving me the opportunity to work with him on very challenging and interesting healthcare management research problems. With his technical expertise, his insightful comments and advice, he helped me to successfully complete this thesis. Dr. Baras' continuous energy and passion for cross-disciplinary research will be a life-long inspiration.

I would like to sincerely thank Professors Jeffrey Herrmann and Guangming Zhang for serving on my committee and for their constructive and meaningful feedback and suggestions during my thesis review examination.

I could not have completed this thesis without the continuous psychological support from my parents, Antonia and Michael and my brother George and my close friends that had a supporting listening in this endeavor.

I would like to express my gratitude for the support received throughout my MS studies and research from National Institute for Standards and Technology (NIST) grant 70NANB11H148 and the Lockheed Martin Chair in Systems Engineering.

These two years I met and built friendships with exceptional people, whom I would like to thank for their cooperation and interesting discussions on research.

Finally, I would like to thank Mrs. Kim Edwards for her untired and efficient manner in handling all the administrative issues, for being there when I needed help,

and the ISR staff for doing their best to help students with their official and academic matters.

.

Table of Contents

This Table of Contents is automatically generated by MS Word, linked to the Heading formats used within the Chapter text.

List of Tables

List of Figures

Chapter 1: **Introduction and Problems Addressed**

Section 1.1: Motivation and Significance of the Problems

Healthcare, in all its forms and systems, is currently a major challenge for the whole world and for industrialized countries alike [1-7]. Healthcare systems around the world are facing unprecedented challenges. The issues are truly global [2-5]. Healthcare costs are rapidly increasing (rose 2.6% in 2013, accelerating to an average of 5.3% per year over 2014-2017) as years go by and unfortunately coverage and offered services are decreasing. Health care, among both providers and payers in public and private settings, is a very costly industry sector. The Economist Intelligence Unit (EIU) estimates [2] that global health care spending as a percentage of Gross Domestic Product (GDP) will average 10.5 percent in 2014 (unchanged from 2013), with regional percentages of 17.4 percent in North America, 10.7 percent in Western Europe, 8.0 percent in Latin America, 6.6 percent in Asia/Australasia, and 6.4 percent in the Middle East/Africa. Among developed nations, health is the second-largest category of government spending, after social protection (social assistance, health/unemployment insurance).

Although an important reason for these trends is the increasing population of senior people throughout the world, there are several other important reasons

tied to fundamental inefficiencies and unnecessary costs and efforts in all aspects of healthcare delivery systems [1-7]. Life expectancy is projected to increase from an estimated 72.6 years in 2012 to 73.7 years by 2017, bringing the number of people over age 65 to around 560 million worldwide, or more than 10 percent of the total global population. In Western Europe the proportion will hit 20%; in Japan, 27%. The aging population will create additional demand for health care services in 2014 and beyond. With aging populations, an increase in those inflicted with chronic ailments that require more health care spending, government initiatives to increase the access to care in both industrialized and emerging markets, and treatment advancements expected to drive sector expansion, pressure to reduce health care costs remains and is escalating. Health care cost increases can be attributed to numerous factors [2] including: healthcare industry consolidation, prolonged hospital stays, rise of expensive complex biologics, overuse of medical services. High healthcare costs are adversely impacting patients as well as providers and insurers. Unfortunately, higher costs do not necessarily correlate to better results or higher-quality care, even in developed countries [2]. Heavy government debts and constraints on tax revenue, combined with the pressures of aging populations, are forcing health payers to make difficult decisions on benefit levels.

There are four major issues that governments, health care providers, payers, and consumers face in 2014: aging population and chronic diseases; cost and quality; access to care; and technology. A key shared demographic trend creating increased health care demand is the spread of **chronic diseases** [2] --

heart disease, stroke, cancer, chronic respiratory diseases, **diabetes**, and mental illness, among others -- which is attributable to the aging population, more sedentary lifestyles, diet changes, and rising obesity levels, as well as improved diagnostics. Chronic diseases are, by far, the leading cause of mortality in the world, representing 63 percent of all deaths. Cancer and heart disease are becoming major killers, even in emerging markets. Africa, the Middle East, Asia, and Latin America are experiencing epidemics in diabetes and cardiovascular illnesses. China, with 92 million diabetics, has overtaken India (80 million) as the world leader in diabetes cases, according to International Diabetes Federation [2]. The cost of treatment for diabetes and other chronic diseases, which may be out of reach for many consumers, especially in emerging markets, is expected to compel a more intense focus on disease education and prevention by governments and health care practitioners while life sciences companies continue to develop innovative new medicines to address many of these diseases.

Improving health care access is a major goal of governments around the world, and a centerpiece of many reform efforts [2]. In the United States, for example, the Congressional Budget Office (CBO) has estimated that, by 2020, approximately 24 million people will purchase coverage through the new federal and state health insurance exchanges established, a substantial addition to the market. To expand citizens' access to medicine in India, the government in 2012 allocated $5.4 billion under a policy to provide free generic drugs/products for patients in government hospitals and rural clinics. While facilitating increased

health care access is an important and worthy endeavor, more people in the system means more demand for services that numerous health care systems are unable to accommodate due to workforce shortages, patient locations, and infrastructure limitations, in addition to the cost issues identified earlier [2]. Many countries across the globe are facing a challenge to meet their required number of health care workers, a shortage that directly affects the quality of care. Globally, the number of doctors per 1,000 population is expected to remain virtually the same between 2012 and 2020 [2-5].

More than one billion people worldwide lack access to a health care system [2-5], both for caregivers and facilities. The United Kingdom, for example, had an estimated shortage of 40,000 nurses in 2012, and has a shortage of other health care professionals, including general practitioners (GPs) [2]. According to a European commission, there will be a shortage of 230,000 physicians across the continent in the near future [2]. The number of caregivers in 36 countries in Africa is inadequate to deliver even the most basic immunization and maternal health services. Rapid economic development across Asia has led to hugely increased access to health care, yet coverage across the region remains uneven. Uneven distribution of caregivers is also a problem.

Bolstering the number of professional medical, nursing, and other health care professionals is not the only staffing challenge facing hospitals and health systems in 2014 and beyond: Organizations will need to source, recruit, and retain staff, such as advanced nurse practitioners and telemedicine technicians,

who are trained to meet the needs of new 21st-century health care models. Workforce shortages are a major contributor to health care access problems around the world; patient location can be another deterrent to care. In many countries, about 80 percent of the population lives in rural areas [2-5]. Many of these rural areas lack good hospitals when compared to urban areas. Finding innovative solutions to provide health care outside of the traditional hospital setting is going to be critical for industry stakeholders. A third constraint on patient access is lack of health care infrastructure in certain countries and outdated facilities in both developed and emerging markets. Due to the lack of a primary care infrastructure in many countries, patients go directly to hospitals, raising both costs and hospitalizations rates.

Across the world, health care systems are recognizing the need for innovation; advances in health technologies and data management can help facilitate new diagnostic and treatment options; however, these same advances are likely to increase overall costs, prompting widespread efforts by public and private health care providers and insurers to contain expenditure by restructuring care delivery models and promoting more efficient use of resources.

Health care technology changes will be rapid and, in some parts of the world, disruptive to established health care models [1-7]. Some exciting advancements are taking place at the intersection of information technology and medical technology. In addition, the use of big data and analytics to gain insights is an active industry trend. Providers can leverage vast amounts of patient data

gathered from a variety of sources to determine the clinical value of specific treatments and how to make them better [1-7]. Technology advancements are also connecting developed and emerging markets and participants along the health care value chain. Adoption of new digital health information technologies (HIT) such as electronic medical records (EMRs), telemedicine, mobile health (mHealth) applications, and electronic medical prescriptions is driving change in the way physicians, payers, patients and other sector stakeholders interact [1-7]. These technology-based changes are shifting the power balance within the health care system and driving different dialogues along the value chain.

Health Information Technology (Health IT) [1-7] has great potential and promise to ameliorate these problems, and is being aggressively pursued in the US, Europe and many other countries [1-7]. **Health IT is of central interest and importance for the problems addressed in this thesis**. There are several fundamental reasons supporting this statement. First, electronic health records (EHR) owned by patients can provide invaluable functionality and service through tracking of patient time histories, multi-sensory medical test data and measurements over time, and can facilitate comprehensive and integrative health monitoring, prognostication and management. Second, it is essential for providing information to patients about their health conditions and treatment progress, and thus making patients active partners in their healthcare management for their entire lifetime. Third, it can support in an interoperable manner, a multitude of heterogeneous sensors (including hand-carried mobile wireless, implanted, wearable, etc. and information/data high speed connectivity

between patients, their doctors and medical centers/hospitals. Fourth, it provides several systems that can efficiently track health quality and treatment effectiveness and results and eliminate unnecessary procedures and costs. Fifth, it can facilitate the dialogue between health care providers and patients. Sixth, it can accumulate data and information from millions of patients, cases, treatments, medicines, medical tests, in a richly indexed manner, that can be analyzed to discover trends, successes and failures, medication side effects etc.. Seventh, it can dramatically improve the management and delivery of healthcare by all practitioners, providers, insurance companies, laboratories, medical practices and hospitals, while tracking costs and quality. Eighth, it can support efficient social networks supporting participatory healthcare and relevant knowledge generation, screening and maintenance. Ninth, it can eliminate unnecessary and costly hospital visits, tests, procedures, etc. Tenth, help develop and incorporate learning capabilities in many of these systems, so as to become richer and more useful through the years.

Despite all these great promises and potential, the deployment of Health IT has been very slow and is neither easily acceptable nor becoming an indispensable part of everyone's life, as it should. The problems encountered are very complex and diverse and they involve human behavioral and psychological phenomena, political challenges, regulatory and legal challenges, debates and contentions among the major shareholders who are healthcare providers, health insurance providers, patients, technologists [1-7]. Health IT systems are complex systems and even systems of systems [1] and need to be treated as such. Health care

systems in general are also complex systems and even systems of systems. Therefore it is imperative that are modeled, designed, constructed and operated as systems; that are taking a **holistic and integrative systems view** [1]. The challenge is even greater because humans of various capabilities, functionalities and roles are essential parts of health care systems, and indeed in large numbers and in a heterogeneous involvement. **What has been lacking in these developments**, as emphasized in the recent report to President Obama [1] **is a modern systems engineering approach to the modeling, design, construction, operation and maintenance of such systems**.

This grand challenge provides the main motivation for the work and research reported in this thesis. We are addressing a specific class of health care management systems (HCMS), as a first but important step towards the systematic application of modern **Model-Based System Engineering (MBSE)** methodologies, frameworks and tools for the design, construction, operation and maintenance of such systems. We selected as focus the modeling and management of **Diabetes Mellitus (or Diabetes 2)** as a problem of high impact, because it affects tens of millions of people world-wide [8]:

> "Already, 366 million people have diabetes and another 280 million are at identifiably high risk of developing diabetes. If nothing is done, by 2030 this number is expected to rise to 552 million with diabetes and an additional 398 million people at high risk. Three out of four people with diabetes now live in low-and middle-income countries. Over the next 20 years, Africa, Middle East and South-East Asia regions will shoulder the

greatest increase in diabetes prevalence. Even in rich countries, disadvantaged groups such as indigenous people and ethnic minorities, recent migrants and slum dwellers suffer higher rates of diabetes and its complications. No country, rich or poor, is immune to the epidemic."

In the USA alone in 2012 29.1 million people, or 9.3% of the population, have diabetes, while the associated costs were estimated for 2012 at $245 billion [9, 10, 12, 13]. Furthermore in the USA alone for the same time period 86 million adults (more than 1 out of 3) have pre-diabetes [11, 12, 13].

Section 1.2: Overall Goals and Contributions of the Thesis

The goal of our research was to **describe a methodology and a framework that utilizes recent advances in MBSE and associated tools, to develop a conceptual architecture for such a system for Diabetes 2** with the following characteristics and capabilities, which are justified from well-established needs for such systems [1, 6, 14, 15]:

(i) Is ***scalable*** to millions of patients, and tens of thousands of healthcare providers.

(ii) Is ***expandable***, in the sense that it can continuously accommodate new data and knowledge, new tests, new, models, new treatments.

(iii) Is ***linkable*** to distributed medical databases.

(iv) It has capabilities to ***"learn".***

(v) It can be ***easily used*** by healthcare providers, health insurance managers, and patients.

(vi) It can operate in a ***distributed collaborative manner*** and be linked to extensive communication and data networks and large heterogeneous sensors and databases.

(vii) It ***can provide quantitative answers to "what-if"*** *type of questions* such as: what is the effect of using modern monitoring wearable technology, what is the most effective test, what is the most effective treatment, what are the tradeoffs between costs and tests and treatments.

Our research was further focused on a key component of such a system, which is the Reasoning Engine to perform efficiently the required difficult tradeoffs in many key decisions. It was not our goal to develop such a system that can be immediately used in medical practice. Indeed this is not possible without the collaboration of many medical practitioners and medical components of a healthcare system, most importantly extensive medical record databases. Rather we wanted to demonstrate the potential and capabilities of such methodologies and the value they provide to all involved in the health acer delivery and management, even with synthetic data, tests, and treatments. Clearly we wanted to demonstrate the huge potential of systematically utilizing modern MBSE methods towards reaching the goals and needs described in [1], and encourage the use and development of such systems linked to real-life data over extensive periods of time. Only when this last step is incorporated into such systems, their full potential and value can be appreciated.

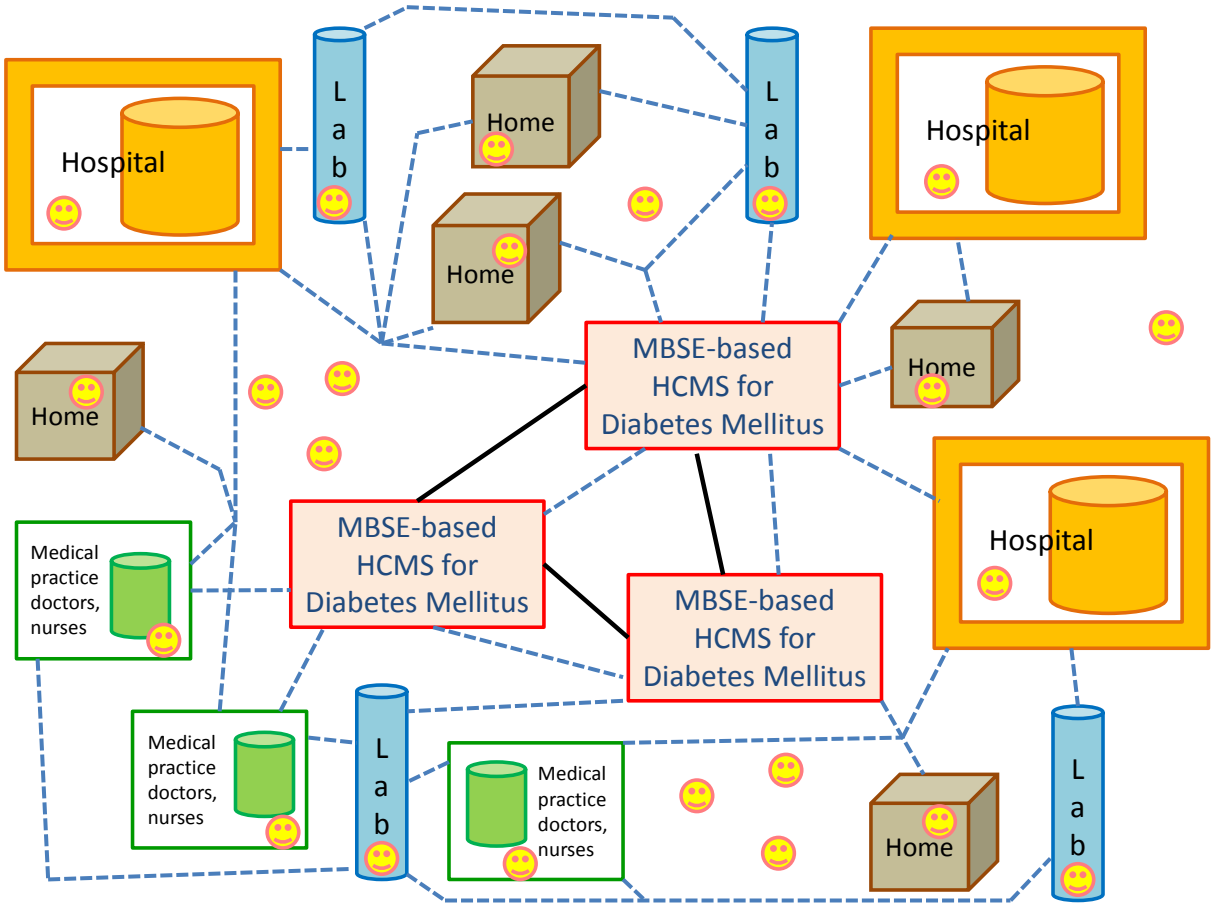


Fig. 1: Illustrating the MBSE-based Health Care Management System for diabetes mellitus and its functional connectivity.

In Figure 1 we illustrate the overall architecture of the type of Health Care Management System (HCMS), based on modern Information Technology (IT), we have in mind and its connectivity to facilities, labs, hospitals, shareholders (patients, doctors, insurance managers, etc.). The dotted lines indicate a modern communication network connecting humans, facilities and medical delivery units. Such networks have been developed and are under development in many countries including the USA [14, 15]. For example in Massachusetts [16] such a "health information highway" has been developed and the majority of hospitals and medical practices are linked via it. In the next phase of its development patients will be linked to it [16].

The proposed system will be accessible, with different accessibility rights and controls, by individual patients, medical doctors, nurses, managers, etc., through either wireless or wireline networks. Indeed the expanding broadband communication infrastructures render the operation of systems like the proposed ones very feasible.

The **contributions of the present thesis** are briefly described below. We developed (Chapters 1, 2 and 8) a framework for Health Care Management Systems (HCMS) of chronic diseases using modern Model-Based System Engineering (MBSE) methodologies and applied it to Diabetes Type 2 (Diabetes Mellitus). Throughout this thesis we focused on Diabetes Mellitus as the driving and focusing application. We described the desired architecture of such systems and the associated connectivity to heterogeneous users, medical facilities, sources of data (Chapters 1, 2 and 8). We also described several high-level characteristics that such systems must have, based on literature review and current and future needs for Healthcare Information Technology and its promise and utility in current and future Healthcare provision and management [1, 6, 14-16]. We developed a disease progression model for Diabetes Mellitus of the Controlled Hidden Markov Chain type, that incorporates diagnostic tests and interventions (Chapters 3 and 4). The model developed incorporates known characteristics and features of the disease from clinical studies and databases, as well as several simulation and experimental models of the disease. We described the diagnostic tests and interventions, including the use of modern Health Care IT devices and systems, used in the diagnosis and treatment of Diabetes Type 2, and developed quantitative models for their operation and evaluation,

appropriate for inclusion in our overall MBSE methodology (Chapters 3 and 4). We incorporated these models to develop a Controlled Markov Chain model for the progression of the Diabetes Type 2 disease, that has three states, incorporates measurements from three diagnostic tests, incorporates actions from ten interventions, and specifically models three types of patients (Chapters 3 and 4). Recognizing the significant role that human behavior plays in healthcare management, we developed models for three types of patients, characterizing their risk profiles through weights representing the value or significance each patient type places on her/his health state (Chapter 5). We developed evaluation metrics for healthcare quality and the associated cost, using these models (Chapter 5). The health quality metric developed combines patient health state counting (occupation) statistics (i.e. number of time periods from a finite management horizon patient's health is in a particular state) for finite time patient health histories and the risk profiles of patients (Chapter 5). We described several other metrics and used some of them in our analysis and experiments (Chapters 6, 7, 8, 9). We developed a Reasoning Engine, an important component of a HCMS, based on tradeoff analysis methods and algorithms, to aid in decision making and analytics (Chapters 5, 6, 7). The novel characteristic of the Reasoning Engine we developed is the linkage of tradeoff analysis methods and algorithms with dynamic disease progression models incorporating tests and interventions (Chapters 4, 5, 6, 7). This contribution is the main contribution of the present thesis. Using these metrics and the disease progression models, we developed and analyzed the performance of two

methods for performing tradeoff analysis between healthcare quality and healthcare cost. The first, is an exhaustive Monte Carlo simulation followed by Pareto point (and frontiers) computations (the EMCS method, Chapter 6), and the second one uses multi-criteria optimization, via deterministic and stochastic Dynamic Programming with full state information, to compute Pareto points (and frontiers) (the FOMCO method, Chapter 7). The second (FOMCO) obtains similar results as the first (EMCS) at a fraction of the time of the first (EMCS). We developed examples of fundamental MBSE constructs (SysML- based) for components of the Reasoning Engine (Chapter 8). We described the decision making and analytics capabilities of the Reasoning Engine, by combining its fundamental tradeoff capabilities with additional statistical computations and considerations, via examples with interesting queries, questions, problems of practical value to Health Care management (Chapter 9).

Section 1.3: Organization of the Thesis

The organization of the present thesis is as follows. In Chapter 1 we introduce the problems of interest and their significance. We describe our overall MBSE based approach for developing a framework for Health Care Management Systems for chronic diseases. In Chapter 2 we provide a description of modern Model Based Systems Engineering methods and constructs. In Chapter 3 we provide an introduction to our disease progression model for Diabetes Mellitus, that incorporates diagnostic tests and interventions, as well as descriptions and models of the tests and interventions that we include in our studies. In Chapter

4 we provide more detailed descriptions of the Controlled Markov Chain disease model we use in the thesis and related analytical models and constructs. In Chapter 5 we develop the analytics and models for the Health Care Quality and Cost metrics. In Chapter 6 we develop the analytics of our first tradeoff analysis method, the EMCS, based on Monte Carlo simulations and subsequent Pareto point computations. In Chapter 7 we develop our second method for tradeoff analysis, the FOMCO, based on optimization via deterministic and stochastic dynamic programming. In Chapter 8 we provide basic MBSE constructs for components of the Reasoning Engine, such as the EMCS and FOMCO methods. In Chapter 9 we provide demonstrations of the decision making and analytics capabilities of the Reasoning Engine developed, via practical examples and problems of interest from a health care perspective. In Chapter 10 we provide conclusions and directions for future research, including the extension of the models and the methods to the more realistic case, where the disease state is estimated from diagnostic tests, that should include quantitative treatment of errors and selection/management of tests. Finally we include two Appendices, with graphs from our extensive simulation experiments. Appendix 1, provides Pareto frontiers for two metrics from 32 simulation runs on 10,000 patients. Appendix 2, provides Pareto frontiers for three metrics from 9 simulation runs on 100,000 patients.

Chapter 2: **Model-Based Systems Engineering – Introduction of our Approach**

Section 2.1: Model-Based Systems Engineering (MBSE)

The term "Model Based System Engineering" could be described as a rigorous, quantitative process for representation of system structure and behavior components to support system requirements management, design, verification and validation activities, beginning with the conceptual design phase and continuing through-out development, operations, and later life cycle phases [17]. In comparison to the document centric approach, in the modern MBSE approach the component models and their interconnections are the main artifact of each procedure and are used for the communication between the dissimilar groups that participate in the system development. In the MBSE setting the information enclosed in the models should be dependable through all the stages. Furthermore, models prerequisites and assumptions (including regions of validity and approximations) need to be established cautiously in order to induce correct problem and design solutions. They must to be precise and sufficient but concurrently avoid enclosing redundant facts that enhance the complexity at no apparent value.

In Figure 2 the essential phases of the MBSE process are indicated [18, 19, 20]. For each system the initial step of the process is the offered evidence. Subsequently, the initial system requirements and the anticipated measures of effectiveness (MoE) are developed (captured). The MoE are used subsequently at the trade-off

phase to guide the selection of components, selection of design parameters and other design space

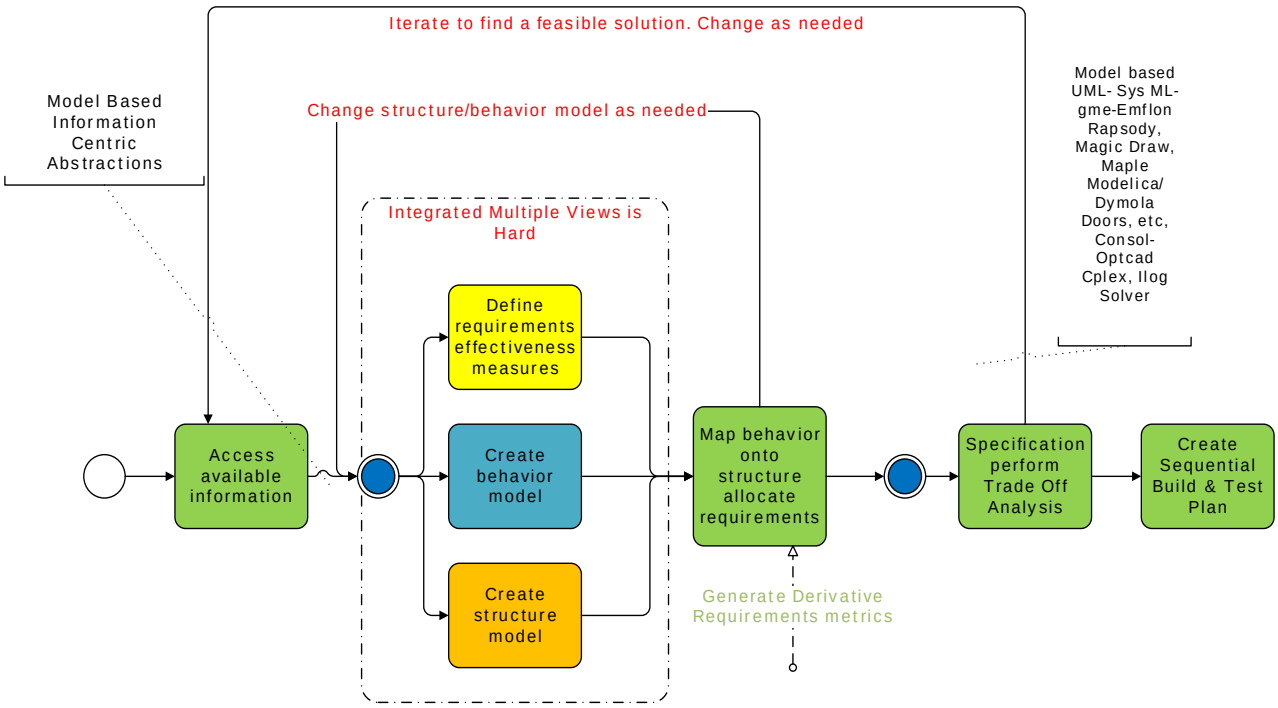


Fig. 2: MBSE process information centric abstractions

exploration functions. After the initial requirements phase, the models of behavior and structure are settled. This includes the creation of components and their connectivity for both the structure and behavior of the system. Subsequently mapping behavior into structure is performed, establishing which components of the structure are involved in each behavior component or set of components. At this stage system architecture has been created. Use cases are systematically used to guide these steps. Throughout the MBSE process derived requirements are produced and thus, if desired, alterations to the system requirement are implemented. Subsequently requirements (both initial and derived) are

allocated to the components of structure and behavior and represented mathematically as either constraints or metrics to be optimized. As already mentioned, the tradeoff analysis phase is used to select the best feasible solution (design, construction, operation) depending on the specified requirements and measures of effectiveness. After exploring the design space and choosing the best alternative the system shall be validated and verified. This stage is critical because it ensures that all the requirements are satisfied and that the system meets its objectives.

Section 2.2: Using System Architecture Model as a System Integration Framework

System architecture is the abstract model that describes the structure, behavior, their components and interconnections, the mapping of behavior onto structure and various views of a system. An architecture narrative is a formal explanation and illustration of a system, organized in a way that supports rationale about the structure and behavior of the system. System architecture can include system components, the outwardly perceptible properties of those components, the relationships (e.g. the behavior) between them. It can deliver a plan from which products can be obtained, and systems developed, that will work together to implement the whole system. There have been efforts to formalize languages to define system architecture; collectively these are called ***architecture description languages***.

Figure 3 shows the essential components of the System Architecture integration framework including hardware models, software models, analysis models, verification, and requirements components. The main challenge and need is to develop scalable holistic methods, models and tools for enterprise level system engineering. Therefore integration of multiple domain modeling tools, trade off tools, validation/verification tools, databases and libraries annotated and component models from all disciplines is required. The benefits from the proposed methodology include broader exploration of the design space, modularity, flexibility and agility. The engineering tools that will be used allow conceptual design to lead to full product models with easy to implement modifications that are also traceable. Last but not least, the proposed methodology enables validation/verification integration with design space exploration tools, through the aid of SysML integrated models. These SysML-based integrated modeling hubs

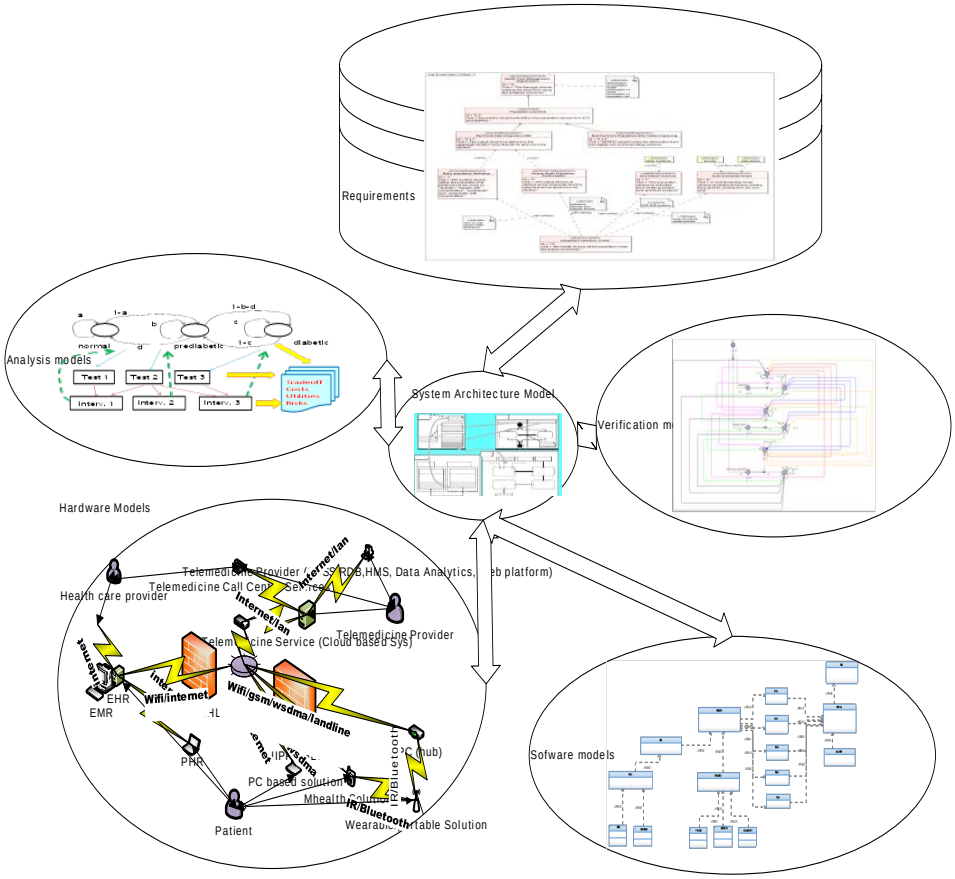


Fig. 3: System architecture integration framework for a healthcare decision support system

(system models) enable system and component model transformations (metamodels), and also linkage to efficient industrial strength tradeoff analysis tools across different domains. Such methods and tools have been developed, used and demonstrated recently (last five years), for many engineering and technology domains, including manufacturing. They have inspired our research to develop similar system models and processes for healthcare support and management systems.

Section 2.3: **Tradeoff Analysis via Multi-objective Optimization and Decision Making**

The design and operation of large complex systems is always guided by complex tradeoffs between numerous nonlinear objects and the need to satisfy many constraints. Multiple objectives occur because a virtuous design balances the attributes of economy, performance, reliability and quality. The resolution of multi-objective optimization problems can be stimulating on a number of facades. One challenge is design objectives and constraints to be optimized subject to statistical variations (uncertainties). For engineering systems defined by large set of differential equations, function evaluation can be very lavish.

Engineering systems are typically designed to satisfy the needs of multiple stakeholders needs. Each stake holder will have:

- a set of functional requirements
- levels of performance that need to be met and
- finally a budget.

Satisfying all of these (often conflicting) criteria typically results in tradeoffs. Figure 4 shows tradeoff curves in multidimensional design space. For a fixed overall system performance, an increase in one intervention therapy (in the case of diabetes care management) typically causes decrease in another factor (metric, objective).

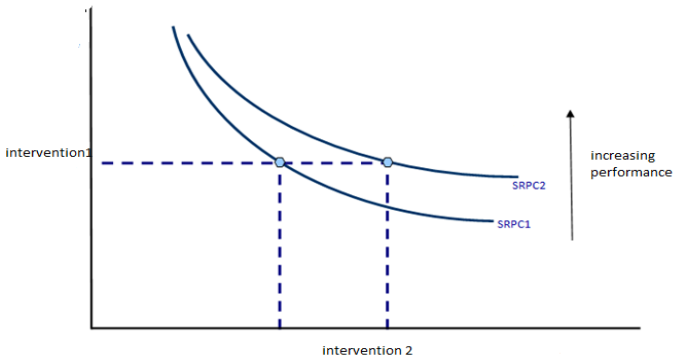


Fig. 4: Tradeoff curves in multidimensional (multiple objectives) design space

For example, in the case of diabetes care management:

1. More functionality usually means less economy (more interventions increase health but also increase cost);
2. Improved performance usually means less economy (interventions that cure timely tend to be more expensive);
3. For systems having a fixed cost, improvements in one aspect of performance may only be possible with a decrease in other aspects of performance.

Tradeoffs also appear in decisions on the use of resources details and timing of implementation, for example:

1. Serial versus parallel implementations;
2. Speed of system implementation versus cost;

Smart systems have a model that describes the relationships among the components of the system. When changes in the implementation factors can be

managed in an orderly way performance of the system can be enhanced. Also a smart system should have the ability to predict and anticipate events.

Optimization could be defined as a process that systematically takes the form of finding the values of design variables "***X***" that maximize an objective function. Optimization algorithms receive as their input information on "***X***" the system inputs and outputs (I/O), the problem goals, and generate a revised set of decision variables "***X_new***." Optimization techniques include:

- Simple trial and error search strategies
- Mathematical programming techniques
- Search programming techniques
- Constraint based reasoning techniques

Multi-objective optimization problems deal with "optimization" of two or more objectives. They search for an *r*-dimensional value vector that provides a good compromise for several objective functions

$$\textbf{\textit{Objective}} = [\, f_1(x),\, f_2(x), \ldots,\, f_r(x)] \qquad (1)$$

and an *m*-dimensional solution vector *X* that is feasible. Often the term "vector optimization" is used for such problems [21], since one can form a vector of objectives like in equation (1).

Weighted formulation or ***scalarization*** [21] is the most straightforward way of handling such multi-objective problems, by converting them to a single objective function via a weighted sum:

$$f(X) = \sum_{i=1}^{r} w_i f_i(\mathrm{X}) \qquad (2)$$

,

where $w_i > 0$ can be thought of as giving the relative importance of the individual objectives in optimizing $f(X)$. With this equation in hand one can compute a solution to the original problem using an algorithm developed for single objective optimization. For this approach to work well, the coefficients w_i must take into account the different scales of values of the objectives in $f(X)$ and the various types of units associated with the objectives. Furthermore great care has to be exercised in properly normalizing the objectives to avoid erroneous tradeoffs due to vastly different dynamic ranges in the values of the various objectives. Another straightforward method, but more expensive computationally, is to select one objective as the one to optimize and treat the others as constraints [21]. By varying the constraints various ***tradeoff points*** can be computed (called ***Pareto points***) [21].

The main difficulty with multi-objective programming problems is computing an "optimal" solution, especially, when the objectives are truly different and conflicting. How for example should one balance project economic concerns with intangibles such as health performance and reward for following a specific intervention? This problem can be solved in part by redefining the principal goal of multi-objective programming. Instead of trying to compute a single optimal solution we implement a solution procedure that partitions the feasible design space into regions of high technical efficiency and regions of inferior performance. The preferred designs are the ones that are technically efficient. The inferior solutions are removed from further consideration.

Chapter 3: **Introduction to Diabetes Mellitus Disease Models, Tests, Therapies**

Section 3.1: Overview of Diabetes Mellitus and Disease Models

Diabetes mellitus is a metabolic syndrome characterized by chronic hyperglycemia due to insulin deficiency, insulin resistance or both [8, 9, 22-26]. Diabetes is a chronic illness that requires long-term continuing medical care and patient self-management education in order to reduce the risk of acute complications. Type 2 Diabetes is a chronic disease with long term complications such as blindness, renal failure and increased risk for stroke and myocardial infraction [22 - 28]. There are several models and algorithms [29] for predicting Diabetes risk and progression based on clinical data. Popular models that are widely used for prediction of diabetes include IRIC [27], QDScore [30], DESIR [31] and UKPDS [32] and can simulate the progression of risk given some vital signs. The model we developed and utilize is influenced from several models such as Archimedes [33-34], Michigan Model of Diabetes [35], UKPDS [32] and DESIR [31] Diabetes risk score models. We are incorporating the advantages from every model and we integrate them into a new system model that is more complete and detailed.

The Archimedes model [33, 34, 36] follows an object oriented approach, differential equations and features. The Michigan model [35] specializes in the health complications induced (by Diabetes) into other vital organs and systems

of humans. UKPDS [32, 37- 41] utilizes an outcomes methodology to evaluate risks of complications in other human physical systems. The main health loss, if one could quantify the health conditions of someone that is diabetic, is the loss from subsequent health complications like renal or retinal problems. The DESIR score incorporates metabolomics tests for predictive diagnosis and focuses on evaluating risks for developing diabetes or for progressing to worse diabetic cases [31, 42, 43, 44].

We posit that Diabetes II disease progression and management can be modeled by a finite state automaton or a Markov Chain [45- 52]. The Markov Chain is a dynamic stochastic model [45-52], that can be time dependent or time independent, that can represent uncertainties and can be combined with finite state machines; an essential component in SE theory.

In this type of models, every state of the disease is characterized by certain test measurements, associated medical diagnosis, any interventions (therapies) applied, and models and knowledge about the disease progression. For each patient, there is a probability associated with reaching a particular disease state conditioned on the previous state of the patient and other characteristics of the patient (like age, weight, sex, eating habits, etc.). These probabilities are computed based on historical and empirical disease databases of patient data, as well as on the basis of dynamical models governing disease progression. Each patient, at a given state (including healthy ones) can be given a probability of developing the disease (at some state) or basic (or additional) complications.

These probabilities are computed based on historical and empirical disease databases of patient data, as well as on the basis of dynamical models governing disease progression. Coupled Markov Chain models could be used given the mixture of cost, risk and health performance in every step of disease progression in time. They are also used to represent probabilities of complications to other human vital systems resulting from diabetes [8-12, 22-26, 27- 44]. Their parameters are estimated from databases [55-59] and from disease knowledge models [27-44, 63-76].

Section 3.2: Typical Tests for Diagnosing Diabetes Mellitus (Type 2 Diabetes)

For diagnosing of diabetes mellitus there are three main diagnostic tests [22-26]. First we describe these tests briefly below.

A1C Test

The A1C test is used to detect Type 2 diabetes and Pre-diabetes but is not recommended for diagnosis of type 1 diabetes or gestational diabetes. The A1C test is a blood test that reflects the average of a person's blood glucose levels over the past 3 months and does not show daily fluctuations. The A1C test is more convenient for patients than the traditional glucose tests because it does not require fasting and can be performed at any time of the day.

Fasting Plasma Glucose (FPG)Test

The FPG test is used to detect Diabetes and Pre-diabetes. The FPG test has been the most common test used for diagnosing diabetes because it is more convenient than the OGTT and less expensive. The FPG test measures blood glucose in a person who has fasted for at least 8 hours and is most reliable when given in the morning.

Oral Glucose Tolerance Test (OGTT)

The OGTT can be used to diagnose Diabetes, Pre-diabetes, and Gestational Diabetes. Research has shown that the OGTT is more sensitive than the FPG test, but it is less convenient to administer. When used to test for Diabetes or Pre-diabetes, the OGTT measures blood glucose after a person fasts for at least 8 hours and 2 hours after the person drinks a liquid containing 75 grams of glucose dissolved in water.

The levels and the range of measurements in these three tests, that characterize each state of the disease progression model, are given from the following Table.

State	**A1C (percent)**	**Fating Plasma Glucose (*mg/dl*)**	**Oral Glucose Tolerance Test (*mg/dl*)**
Diabetic	> 6.5	> 126	> 200
Pre-diabetic	5.7 to 6.4	100 to 125	140 to 199
Healthy	< 5.7	99 or below	139 or below

Table 1: Defining diabetes 2 disease states from test outcomes
(Definition *mg* = milligram, *dl* = deciliter)

For all three tests, with the Pre-diabetic range the higher the test score the greater the risk of diabetes. The source for Table 1 was adapted from [89] ("American Diabetes Association Standards of Medical Care in Diabetes-2012," *Diabetes Care 2012 (Supp 1): S12*, Table 2).

Section 3.3: Progression of Diabetes States as a Markov Chain

The Markov chain model (see Figure 5) [45-52] represents the probability to move from one disease state to another given your current state and other inputs, or features. A Markov chain model incorporates statistics in different states and state transitions and compared to decision trees, these models have more compactness in representing historical data that are important, and their time evolution. Such a model is also dynamic. States can have values (parametric), or activities/functionalities (0, 1). The time component offers an advantage over decision trees. The Markov chain models could be coupled, and also have output probabilities linking diabetic states and interventions to complications in other human vital systems or improvements (via risks for example).

The major advantage of introducing Markov chain models for the problem of interest in this thesis is that it affords the capability to measure the effects of every possible intervention sequence via various metrics:

i. ***Cost*** (of the disease, and various tests and interventions);
ii. ***Risk*** (of developing diabetes, or for progressing in worse states, and risks

of developing complications in other human systems such as coronary, vision, neurological)); and

iii. ***Utility*** (of various tests and interventions).

There are sequences of random variables in which the future variable is determined by the present variable but is independent of the way in which the present state arose from its predecessors. Markov chain analysis looks at a sequence of events and analyses the tendency of one event to be followed by another. The model can handle costs and outcomes. In every state we can assign weights for the cost and outcome quantities to be estimated. The model can accommodate utilities of tests and interventions; single or sequential (interrelated).

The model that we used to represent disease progression in Figure 5, is a simple model; the states of diabetes disease are: **normal**, **pre-diabetic**, **diabetic**. The probabilities a, b, c, d, are estimated (or can be estimated) from clinical data [8-12, 22-26, 55-59, 63-65] and maintained simulation models of the disease [27-44]. Such

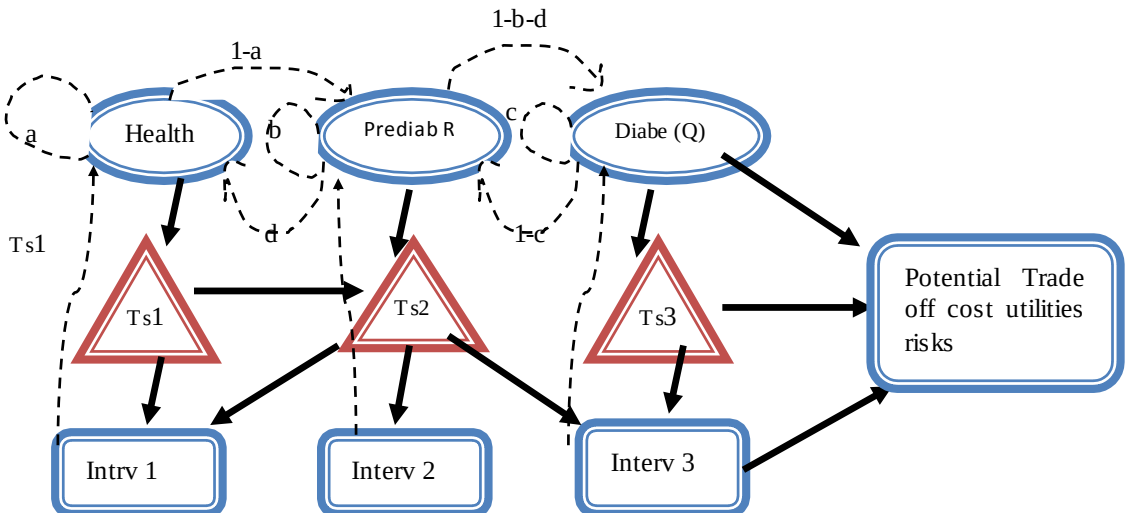


Fig. 5: Example of progression of Diabetes state as a Markov Chain

models can have additional hidden states to capture more complicated aspects and influences of the disease [45-52, 67]. Q and R designate probabilities or risks for damage in other vital systems (outcomes). We model single tests and interventions or sequences of the same costs of tests and interventions, as well as risks and utilities as functions of disease states and outcomes.

From our perspective the most important properties and capabilities of Markov chain type models for disease progression, including hidden Markov (chain) models, controlled Markov chain and controlled hidden Markov models (see Sections XCH of this thesis) are the following. First, the parameters for these models can be ***learned*** from the various data sources indicated just above [8-12, 22-26, 55-59, 63-65, 27-44], and this can be accomplished in an incremental and ***distributed collaborative*** manner once the architecture (with its connectivities) described in Figure 1 is implemented. Second, this type of models accommodates ***expansion***, in the sense that it can continuously accommodate new data as well

as new interventions and tests. Third, hierarchical such models, and their associated parameter estimation and control algorithms are well established in various fields of Engineering and Information Technologies [45-52], and therefore such models are ***scalable*** to very large number of cases and users. These considerations are relevant for the desired characteristics and capabilities of health care support and management systems described in Section 1.2.

Section 3.4: Disease Management Model

The typical disease management model that we developed and used in this thesis is influenced from several models such as Archimedes, Michigan Model of Diabetes, UKPDS and DESIR Diabetes risk score models [27- 44]. We are incorporating the advantages from every model and we integrate them into a new system model that is more complete and detailed. ***An important and innovative aspect of our modeling approach is the linking of the models with trade-off analysis algorithms***, in order to employ the model in trade-off analyses between risk, utility (quality of life), and costs. ***Another important and innovative aspect of our disease progression models is that they are designed to easily accommodate (user specified) finite time horizons for disease management, monitoring and decision making***, vs the more traditionally used and typical models that focus only on long term and life time effects [27-44].

Coupled Markov Chain models could also be used given the mixture of cost, risk

and health performance in every step of disease progression in time. They could also be used to represent probabilities of complications to other human vital systems resulting from diabetes. Their parameters are estimated from databases [55-59] and from disease knowledge models [27- 46]. Such models can lead to enormous numbers of states for a large population of patients. And this fact makes computing complex, and a factor to be considered, when developing data and management analytics for such systems. We have developed methods to handle this type of complexity in our earlier work [53, 54]. For example for N patients we have N^3 states, corresponding to the normal/pre-diabetic/diabetic status of each patient.

Figure 6 illustrates the idea of incorporating a Markov Decision Process (MDP) [45-52] in the management of Diabetes Mellitus.

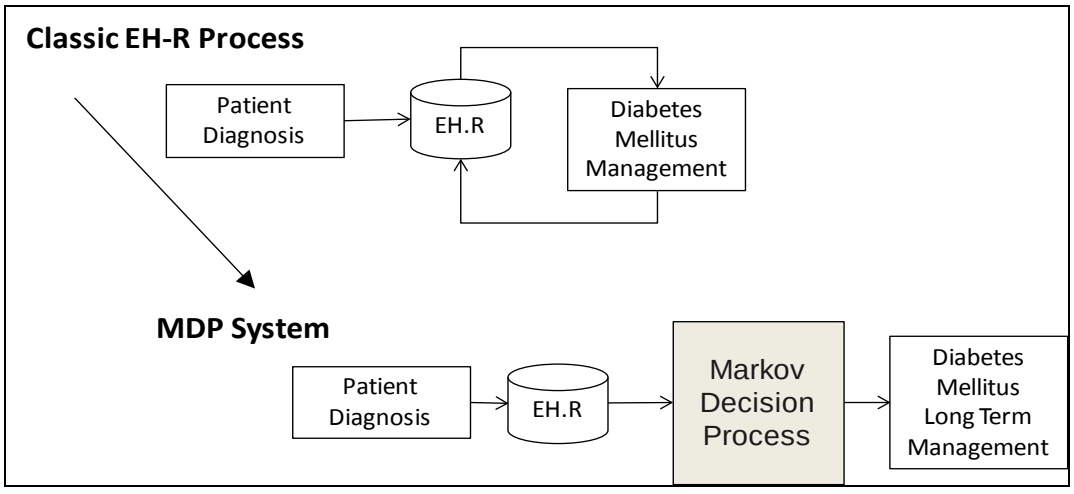


Fig. 6: High level view of the state model for long term management of Type 2 diabetes

A Markov Decision Process (MDP) [45, 47, 48, 60, 83] is a discrete time stochastic control process that provides a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision maker. In MDPs the transition probabilities depend on the controls exercised. MDPs are useful for studying a wide range of optimization problems solved via dynamic programming and reinforcement learning [45, 47, 48, 60, 83]. MDPs were known at least as early as the 1950s (cf. Bellman 1957) [62]. A core body of research on MDPs resulted from Ronald A. Howard's book published in 1960, “Dynamic Programming and Markov Processes” [61]. They are used in a wide variety of disciplines, including robotics, automated control, economics, and manufacturing. Figure 7 illustrates the operation of an MDP model with its controls (actions, interventions) and state transitions.

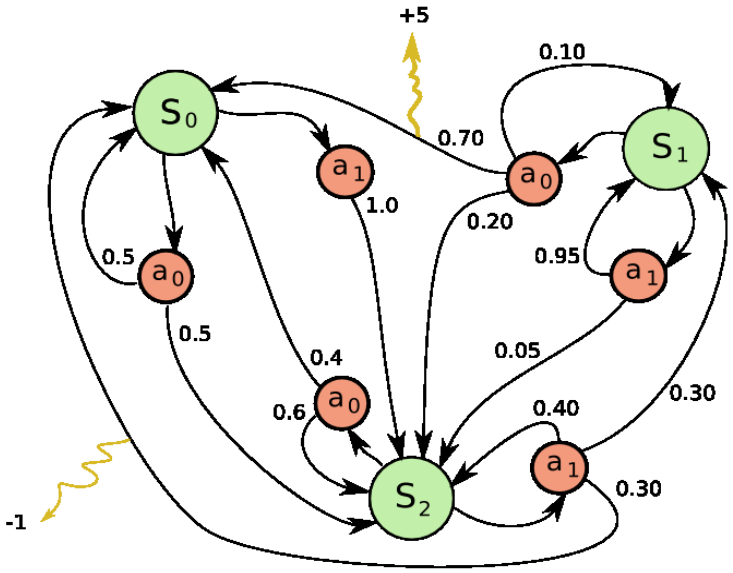


Fig. 7 : Example of a simple MDP with three states and two actions [47]

Section 3.5: Interventions and Therapies Based on Information and Communication Technologies

One of the objectives of our research in this thesis was to develop a framework that will allow the evaluation of the impact of Information and Communication Technologies (ICT) on the diagnosis, treatment and management of Type 2 Diabetes. These technologies include wearable sensors, telemedicine, easy communication between patients and doctors, etc. We describe below some of these technologies.

Subsection 3.5.1: ***EHR***

Electronic health records (EHRs) have the potential to improve quality and safety because they provide better access to information, more reliable

communication between providers, and clinical decision support. According to Propp [77], the implementation of EHRs creates a safe, efficient, and easily accessible system in a healthcare setting. Propp discusses the change from a paper-based infrastructure to an EHR in an Emergency Department [77]. He found that the nursing staff had easy access to the physician's clinical notes and assessments [77]. In the Emergency Department setting, this is a major positive. With easier access to information, nurses will be able to move patients quickly and effectively. This will increase the amount of beds in the hospital and increase the number of patients that can be seen.

Not only does information become more accessible, but also communication between physicians and the staff, patients, laboratories, and pharmacies becomes more efficient and effective. Propp [77] also suggests that physicians were able to provide easily accessible non-verbal task communications with the staff. This is beneficial in a healthcare setting, especially in an emergency department. Instead of waiting on a busy physician to come by with instructions, communication becomes faster and more efficient. This will also improve the quality of care, because physicians will have more time to focus on patients' needs. In addition, King et al. [78] describe how communication between other factors increases with the implementation of an EHR. EHRs can be used to exchange important information with pharmacies, communicate with laboratories and incorporate Lab results into the EHR as structured data, and provide patients the ability to view, download, and transmit their health information [78]. Quality of care increases once other healthcare facilities are

able to enhance communication with physicians.

Electronic health records can also support clinical team interventions when coming to decisions on a patient. A study done by Bates et al. (1998) [79] helps demonstrate how physicians' computer order entries (POE) decrease adverse drug events when used in combination with a clinical team. Results showed that a POE system prevented more than half of the serious medication errors [79]. This POE system was able to reduce medication errors because it included a dose selection menu, simple drug allergy and drug-drug checking, and the requirement that clinicians indicate the route and frequency of drug doses. Furthermore, a computer system resolves the difficulty of translating illegible orders and greatly reduced the need for transcription [79]. EHRs help physicians and their teams to make the appropriate decisions on their patients, further preventing harm and increasing quality of care.

Subsection 3.5.2: ***Telemedicine and diabetes***

Telemedicine is defined as the medical activity that involves an element of distance and use of telecommunication resources and strategy. One of the most well-known applications of telemedicine is the support of reporting and/or screening and then interpreting blood glucose measurements. A common example of the use of these evaluations for these applications is the self-management of other chronic diseases [71].

The benefits of computerized management of diabetes are mentioned and pointed out, for their possible advantages of monitoring and communicating with patients from a distance, in [69-71]. At the same time, these interventions are cost effective means for delivering health care. For the advantages that we have mentioned we reviewed the studies of telemedicine in diabetes care from several quality reviews that have been conducted in the past. We performed a metadata review analysis and we tried to research what is considered quality in telemedicine and diabetes care, how it is captured and measured. We have incorporated these findings in the models and analytical methods and results reported in this thesis.

Chapter 4: **Controlled Hidden Markov Model of Diabetes Mellitus Disease Evolution with Interventions and Tests**

Section 4.1: Mathematical Description of the Basic Model

In the model we have developed we represent diabetes progression as a Controlled Hidden Markov Chain (CHMC). The model has three states for diabetes mellitus, but more complex models can be developed [45-52] based on more detailed dynamics and progression of the disease [8-13, 22-26], supported

by clinical data [55-59] and detailed biochemical models of the disease [27-44, 63-76, 81]. Thus, as already emphasized in Chapter 1 of this thesis, our Model-Based Systems Engineering (MBSE) approach enables incremental improvement of the disease models, their parameters, the tests, the interventions, and the knowledge captured in our healthcare management and evaluation framework. These "learning" and "updating" capabilities are implied by the architecture depicted in Figure 1 (page 10) and by its implementation including modern broadband telecommunications and data linkages to the heterogeneous users and medical databases and facilities. We denote by n_x the number of states in our disease model; for this thesis $n_x = 3$. The states we use have the following interpretation: ***State 1*** represents the **Healthy** (disease free) condition of a generic patient. ***State 2*** represents the **Pre-diabetic** condition of a generic patient. ***State 3*** represents the **Diabetic** condition of a generic patient. In practice these states are determined by a variety of medical tests and measurements and are described in the current medical diagnostic practice for this disease [22-26]. As the patient goes on with her/his life, various tests are performed periodically to determine the state of health of the patient regarding this particular disease, and various interventions (treatments) are recommended and followed, depending on both the test outcomes (measurements) and the state of health of the specific patient. ***In practice, we have to be very careful in discussing state of the disease, because rigorously speaking such an absolute description of the state of the disease does not exist in medicine. Rather, in medicine and healthcare, the concept of state***

represents the collection of ranges in the outcomes of the various diagnostic tests performed. That is the main reason that the most practical type of model we can develop should be of the Hidden Markov Chain type. Our framework can incorporate such models as well. As these models and their analysis are quite complex, we develop and analyze simpler models first as stepping stones towards these more powerful and more realistic models, which (Hidden Markov Chain type models) we plan to develop and investigate in our future research (see our recent paper [87] for our initial work and results in this direction). Furthermore, another complexity that needs to be incorporated is the dependencies of the tests on the past history of the disease with a specific patient, and on the history of tests and interventions performed on the specific patient. This is typically the practice in medicine and healthcare, but unfortunately at this stage of development the Information Technology (IT) support for such considerations of past histories is not implemented, except for very few programs and even there with limited scope. Currently such considerations are performed by the human medical practitioners with minimal support of advanced IT support and tools. This situation though is rapidly improving in the USA and world-wide thanks to the massive investments in Health IT and associated systems and frameworks. The work in this thesis is a contribution in this promising direction.

The purpose of the relatively simplified model, developed and employed in this thesis, of these complex interrelated processes and disease evolution is to allow systematic quantitative studies of the effects of medical tests, called simply ***tests***

from now on, and various treatments, called from now on ***interventions***, on the disease progression. We include as interventions the use of current IT technology such as wearable sensors, smart phones, Internet and cloud-based information services, linkage with hospital, private medical practice and social medical networks over the Internet. Additional technologies can be easily incorporated in our framework. Each test and intervention affect the transitions from one state of the model to the other, and also have costs associated with their use/application. An important objective of the model developed and the study undertaken in this thesis, is to also ***develop a systematic methodology for evaluating the quality of health care provided to a generic patient (meaning the sequence of tests and interventions applied), the associated costs, associated tradeoffs and many other important evaluations of individual tests and interventions, over a particular finite time period.*** The purpose of our study is not to provide a specific medical model with all its complexities and ramifications, but rather develop, use and demonstrate the significant benefits of a MBSE framework for analyzing these problems, which is expandable and scalable to much larger and much more complex models thanks to the industrial strength tools we use in the quantitative analysis. Furthermore, as already emphasized in Chapter 1, our MBSE framework is by construction **modular** and **composable** allowing many tools and models to be easily integrated within its structure. This is in great contrast with the current state of the art in Healthcare IT systems, even with the most advanced ones.

Our analysis specifies a time horizon for the study (or use of the support system developed) denoted by T; in most of our simulation experiments and example cases we take T to be ten years. This time horizon is defined by the user of the framework: a medical practitioner, a medical doctor, a patient, a manager in a medical provider, a manager in a medical insurance provider, a technologist, a policy maker, a regulator, etc.. This horizon can be rolling, allowing extensions of the study and tracking of patients incrementally over longer periods. To simplify matters we develop a discrete time dynamical model, where tests and interventions occur periodically with period

Δ. Thus there are $N_{T,\Delta}$ time steps in our time histories, where $N_{T,\Delta} = N / \Delta$, an

integer. In most of our studies we have taken Δ to be one year. As usual in a CHMC

model, the dynamic evolution is prescribed by transition probabilities, which in our

model depend on the tests applied at the particular time, denoted by μ(t), the

interventions applied at each particular time, denoted by u(t), and various exogenous or parametric factors such as the age of the patient, the type of work she/he does, the environment where the patient leaves, the age of the patient etc., denoted by w(t). The

variables μ, u, w can be scalar-valued, in the case of a single test, single intervention,

single exogenous factor, or vector-valued, in the case of several tests, several interventions, several exogenous factors. In this thesis the type of these variables (i.e. whether they are scalar or vector) will be more or less obvious from the context in each case of their use.

As shown in the diagrammatic representation of our model in Figure 8 there are transition probabilities from each state to the next, as time evolves in steps of length Δ, which from now on we take as 1, for notational simplicity (it may mean

1 year (typically), or one six month period, or a quarter of a year, one month, one day, one hour, etc.). As time evolves, and under the influence of each patient's life evolution, the evolution of the disease and the tests-interventions applied at each time, the state moves with certain probabilities to a different state or stays the same. This is a typical model used to capture similar type of problems in Engineering and Operations Research, where time is discrete, measurements and controls are discrete valued, meaning one selects a measurement (test, observation, sensor) from a finite set of available measurements, and also selects a control (action, intervention, treatment) from a finite set of available controls. Our System Model is inspired by such models which implement the basic feedback cycle of **Sense-Decide-Actuate** (Act) in Engineering and Operations Research, where in our case the corresponding cycle is translated to **Test (Diagnosing) → Decide → Intervention (Treatment).**

In Fig. 8, for example, P_{12} is the probability of transition from *State 1= Healthy* to *State 2 = Pre-diabetic*. u_k is the intervention applied at that time step, where

$k \in \{1, 2, \ldots, n_u\}$, which means that for a given state we could have different interventions that we can follow each time; a practical requirement from medical and clinical practice [22-26]. μ_l is the test applied at that time step, where $l \in \{1, 2, \ldots, n_\mu\}$, which means that for any given state we could have different (diagnostic) tests that we can apply to the patient; again a very useful feature of our model. The transition probabilities in the models developed and used in this thesis do not depend explicitly on time themselves, but they depend on time implicitly through the dependence of u_k and μ_l on time. However, our overall framework, modeling and mathematical/analytical methods can be easily extended to accommodate explicit time dependence in these transition probabilities.

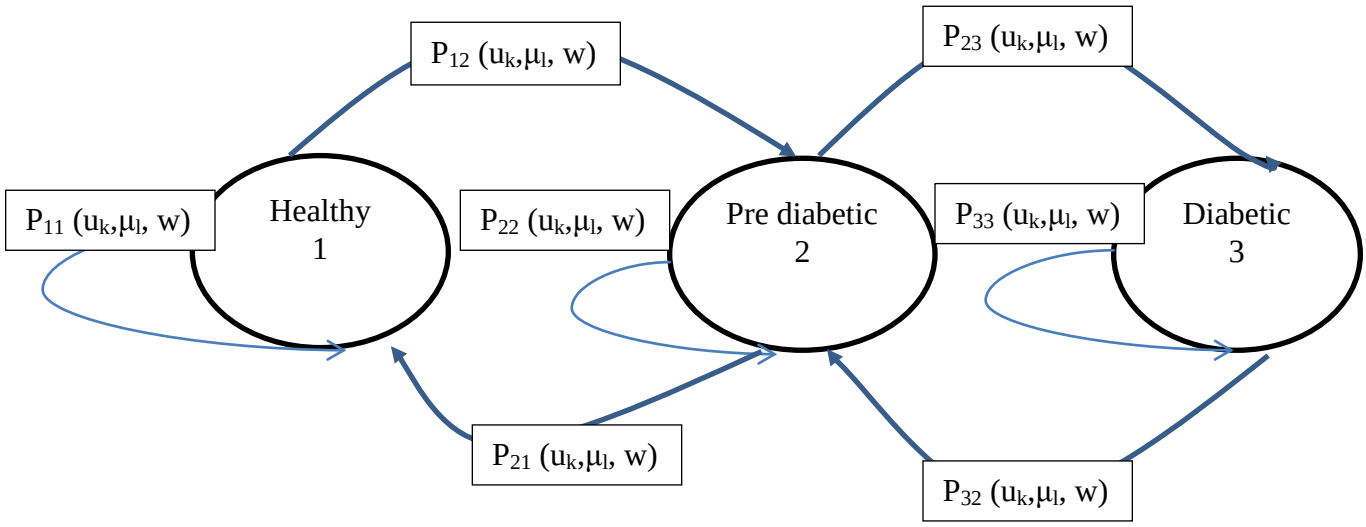


Fig. 8: Graphical representation of the CHMC model developed for Diabetes 2 progression

We also note that in the model depicted in Fig. 8 we do not include all possible state transitions. For example we do not include transitions form state 3 directly to state 1. Rather we allow the state to transition first from state 3 to state 2 and then to state 1. More complex models can be easily accommodated. The models we use are completely flexible on this aspect, and their complexity can be completely defined and supported by the various users. Often, such restrictions come from common sense and from medical and clinical experiences and observations. As already discussed, *w* denotes the various non-controllable factors that influence the dynamic state evolution of diabetes in a generic patient; examples of these are environment, age, gender, ethnicity or family history. We note that in the underlying Markov chain we do not allow transitions form State 1 (Healthy) to State 3 (Diabetic) directly, and reversely, without passing through State 2 (Pre-diabetic) first. This is based on practical medical evidence for the majority of patients. Clearly, if we want to include such direct transitions, the modification required for our model is trivial.

Section 4.2: Operation of the Model, Tests, Interventions, Feedback

In this section we describe in some more detail, the model, the tests, the interventions and various operational scenarios with different degrees of feedback. The main diagnostic tests were described already in Section 3.2 of this thesis.

Subsection 4.2.1: ***Operational characteristics of the diagnostic tests***

As is typical, medical diagnostic tests, measure an important (or several important physical variables) of the patient, and in doing so errors can occur. Typically, medical tests employ a threshold (or several thresholds) based policy (i.e. rule). The policies of common tests are standardized for uniformity among patients and doctors, and in order to provide a stable basis for medical judgments. The diagnostic tests provide a decision, that is typically Boolean (i.e. binary valued) but it may also be ternary valued (i.e. take three possible values). Depending on the complexity of the test, and the description of the disease (i.e. how many states are used to characterize the disease), the decision can take more values as well. Typically a single threshold leads to a binary valued output, while for a ternary valued output two thresholds are needed. To describe briefly these operational characteristics and the various diagnostic errors that can occur let us denote by z the scalar-valued physical variable (like temperature, glucose, density of cholesterol cells in the blood, etc.) that the test is designed to measure; similar descriptions and definitions can be given in the case of simultaneously measuring several variables, in which case z will be a vector-valued variable. Let y be the measurement outcome of test μ, designed to measure the variable z. The outcome y is a function of the variable z, $y = F(z)$, which can be known or unknown, invertible or non-invertible. If it is known and invertible, then for all practical purposes one can take $y = z$.

For a binary decision, let ζ be the threshold for decisions, in this case, as we are

using a single threshold. In such binary tests, the goal is to use the test as evidence for two opposite hypotheses, H_1 meaning the disease is likely present in the patient examined, and H_0 meaning the disease is likely absent in the patient examined. Then the single threshold decision policy is mathematically (and operationally) described as follows:

$$\left.\begin{array}{l}\text{If } \zeta < y, \quad \text{then} \quad H_1 \quad \text{is true} \\ \text{If } y \le \zeta, \quad \text{then} \quad H_0 \quad \text{is true}\end{array}\right\} \text{ Single threshold decision policy.} \qquad (3)$$

There are two types of errors associated with this type of decision policy. One error occurs when the test decides, following this policy, that H_1 is true, while in fact H_0 is true. For obvious reasons this type of error is called ***False Alarm Error***. The other error occurs when the test decides, following this policy that H_0 is true, while in fact H_1 is true. For obvious reasons this type of error is called ***Missed Detection Error***. Thus the performance of such a test is quantitatively described following standard statistical hypothesis testing theory [45, 48, 60, 67, 81, 83, 88] by the two probabilities: The probability of making false alarm errors, denoted by P_{fa}, and by the probability of making missed detection errors, denoted by P_{md}. Obviously 1- P_{md} is the probability of correct detection, typically denoted by P_{d}. In binary classification terminology, often used in medicine [45, 48, 67, 81], P_{d} is also called the ***True Positive Rate***, while P_{fa} is also called the ***False Positive Rate***. In this thesis we use

these terms interchangeably. All these probabilities depend on the threshold ζ, the

statistical descriptions of *z*, *y* and the disease characteristics vs. patients, and more generically on the medical description of the disease, the relation of the disease to the variable *z*, and the relation between the variables *z* and *y*, and on several other hidden

variables possibly. The graph of P_{d} vs. P_{fa}, typically as the threshold ζ varies, characterizes the quality of the test holistically, meaning as both a physical medical instrument as well as a decision rule (policy). The resulting curve is the well-known ***Receiver Operating Characteristic (ROC)***, a term widely used in electrical engineering [83, 88] to evaluate various communication systems and signal processing schemes. It is now widely used in medical tests, psychology, radiology, biometrics, machine learning, data mining, and many other science and technology areas.

We next provide a similar brief description of the operation of a ternary decision test.

Let ζ_{l} be the lower (valued) threshold, and ζ_{h} be the higher (valued) threshold. Then,

keeping everything as above including the fundamental interpretation of testing for two hypotheses H_1 and H_0, Then the two thresholds decision policy is mathematically (and operationally described as follows):

$$\left.\begin{array}{l} \text{If } \zeta_u < y, \quad \text{then} \quad H_1 \quad \text{is true} \\ \text{If } \zeta_l < y \le \zeta_u, \quad \text{then test is ambiguous} \\ \text{If } y \le \zeta_l, \quad \text{then} \quad H_0 \quad \text{is true} \end{array}\right\} \text{ Two thresholds decision policy.} \qquad (4)$$

In this case, with this decision rule, there are more errors that can be made. To define them, let's call the ambiguous decision as a decision for hypothesis $H_{1\text{-}0}$. Namely, one type of error that can occur is when the test decision is that H_1 is true, while in fact $H_{1\text{-}0}$ is true. Another one is when the test decision is that H_1 is true, while in fact H_0 is true. Another one when the test decision is that $H_{1\text{-}0}$, is true, while in fact H_1 is true. Another one is when the test decision is that $H_{1\text{-}0}$ is true, while in fact H_0 is true. Finally the last two types of errors are when the test decision is that H_0 is true while in fact H_1 is true, and when the test decision is that H_0 is true, when in fact $H_{1\text{-}0}$ is true. Like in the binary decision case, these errors are quantitatively characterized by the conditional probabilities

$$\begin{aligned} &P(D=\{H_0=true\}\,|\,\{H_1=true\}), \quad P(D=\{H_{1-0}=true\}\,|\,\{H_1=true\}),\\ &P(D=\{H_0=true\}\,|\,\{H_{1-0}=true\}), \quad P(D=\{H_1=true\}\,|\,\{H_{1-0}=true\}),\\ &P(D=\{H_{1-0}=true\}\,|\,\{H_0=true\}), \quad P(D=\{H_1=true\}\,|\,\{H_0=true\}), \end{aligned} \tag{5}$$

where by D we have denoted the test decision following the two threshold policy described above. Similarly as with the binary case, the performance of the test (i.e. its

quality) can be quantitatively characterized by plotting these six probabilities as

functions of the lower and upper thresholds ζ_l and ζ_u . Clearly these representations, analytics and performance are more complex than in the simple binary case. Briefly one constructs the so-called ***Confusion Matrix***, which in this case is a 3 x 3 matrixΦ,

with elements $\Phi_{ij} = P(D = \{H_i = true\} | \{H_j = true\})$ for i and j, taking the values 1, 0, 1-0. Considering the diagonal dominance [80] of the matrix Φ as a function of

the thresholds ζ_l and ζ_u, provides a quantitative performance metric. Since the

elements Φ_{ij} of the matrix Φ are probabilities, they are nonnegative, i.e. $\Phi_{ij} \geq 0$.

Furthermore since the events $\{H_0 = true\}, \{H_{1-0} = true\}, \{H_1 = true\}$ are complementary,

$$\sum_i \Phi_{ij} = 1, \quad \text{for each } j. \qquad (6)$$

An n x n matrix A is called ***diagonally dominant*** [80] if

$$|A_{ii}| \geq \sum_j |A_{ij}|. \qquad (7)$$

For matrices with nonnegative elements, like Φ, we do not need to use absolute

values in the above definition. A simple measure of diagonal dominance of a diagonally dominant matrix A with nonnegative elements is [80]

$$\rho_{dd} = \sum_i (A_{ii} - \sum_j A_{ij}). \qquad (8)$$

Several other metrics can be constructed [80]. Clearly the larger $\rho_{dd}(\Phi)$ is the better the performance of the two threshold strategy is for this three state problem.

Subsection 4.2.2: ***Accuracy, fallacy of each test, state estimation from test results***

We can easily apply the ideas and constructs of Subsection 4.2.1 to our model, since we have three states, 1, 2, and 3. Indeed the two thresholds are used to define the state of the disease, ***since in medicine and biology due to the enormous complexity of the underlying systems, true states are nearly impossible to define, and instead states are defined via bounds on the values of test measurements*** [22, 25, 26, 45, 55, 67, 75, 76, 81]. Thus for the case of our three state model, using the three tests considered, the states are defined using such thresholds based on the Table 2 below (repeated here for convenience of the reader) from Section 3.2 of this thesis (page 24, Table 1).

State	**A1C (percent)**	**Fasting Plasma Glucose (mg/dl)**	**Oral Glucose Tolerance Test (mg/dl)**
Diabetic	> 6.5	> 126	> 200
Pre-diabetic	5.7 to 6.4	100 to 125	140 to 199
Healthy	< 5.7	99 or below	139 or below

Table 2: Model states defined via the three (diagnostic) tests used in the model. Here mg = milligram, dl = deciliter. For all three tests, with the pre-diabetic range the higher the test score (value) the greater the risk of diabetes. [Source adapted from [89] American Diabetes Association Standards of Medical Care in Diabetes-2012. Diabetes Care 2012 (Supp 1): S12, table 2.]

In modern stochastic systems and stochastic control terminology this is the partially observed problem [60, 83], and is quite more complex than the corresponding fully

observed problem [60, 83]. We have left this extension of our work in this thesis as a future research direction, while some initial results in this direction will appear in our recent paper [87]. For the partially observed case our model becomes a Controlled Hidden Markov Chain (CHMC) [60, 83].

Let us denote by $y_{\mu_l}(t)$ the measurements resulting from the test μ_l, $l = 1, 2, ..., n_\mu$ applied at time t. $y_{\mu_l}(t)$ is a scalar real-valued variable. We will denote by $\hat{x}(t)$ the best estimate of the disease state at time t. In general $\hat{x}(t)$ will be a function of the past test measurements and past interventions. The conditional probability mass functions (pmf) $p_{x|y}(t,i) = \Pr\{x(t) = i \mid z^t\}$ play a very important role [60, 83] in modeling this closest to reality formulation, where z^t denotes the entire history of past tests and interventions, before the intervention $u(t)$ is decided and applied; i.e. $z^t = \{y(0), y(1), ..., y(t), u(0), u(1), ..., u(t-1)\} = (y^t, u^{t-1})$. In the complete formulation of the partially observed case the tests and interventions at each time epoch t, are chosen to minimize some expected value of a metric. It is known [60, 83] that for general CHMC models and general metrics, these optimal tests and interventions at time t, given the past history z^t, are explicit functions of the conditional pmf $p_{x|y}(t,.)$. Furthermore the pmf $p_{x|y}(t,.)$ itself satisfies a dynamic recursive equation [83], known as the nonlinear filtering equation, which captures the updating of the state estimates given new received measurements. The actual functions expressing the relationship between the optimal tests and interventions and the pmf $p_{x|y}(t,.)$, are obtained as solutions of a Dynamic Programming equation [60, 83]. It is these two dynamic equations that make the partially observed case substantially more complex

computationally. We will only address the fully observed model in this thesis. Starting from the ideas and concepts just described we have obtained some initial results for the partially observed case in our forthcoming paper [87].

We have collected below statistical information regarding the diagnostic tests we described in Section 3.2. The A1C test result can be up to 0.5 percent higher or lower than the actual percentage. This means an A1C measured as 7.0 percent could indicate a true A1C anywhere in the range from ~6.5 to 7.5 percent. Health care providers can visit www.ngsp.org to find information about the accuracy of the A1C test used by their laboratory. In the study of Cox (2009) [81] the sensitivity and the specificity of every test are discussed. Table 3 illustrates the decision regions for estimating the disease state following a two threshold policy.

$\hat{x}$ = estimated state	μ_1 = A1C	μ_2 = FPG	μ_3 = OGTT
Diabetic = 3	A_3^1	A_3^2	A_3^3
Pre Diabetic = 2	A_2^1	A_2^2	A_2^3
Healthy = 1	A_1^1	A_1^2	A_1^3

Table 3: Relating test measurement ranges to disease state estimation per medical standards.

Table 4 [81] illustrates the sensitivity and specificity of each of the three tests as a function of the "true" state of the disease.

Diabetes state	**A1C (accuracy, fallacy)**	**FPG (accuracy, fallacy)**	**OGTT (accuracy, fallacy)**
Diabetic	78%, 15%	55%,5%	95%,20%
Pre Diabetic	80%,15%	85%,10%	88%.25%

Healthy	84%, 16%	60%,5%	80%,25%

Table 4: Accuracy and fallacy of the three common Diabetes II tests

From the above tables, and similar data from several sources [22-26, 27-44, 55-59, 63-76, 81], in each cell we can estimate the probability of a test to be accurate, the probability of errors as well as other characteristics like sensitivity and specificity. These statistics are clearly into one-to-one correspondence with the statistics and probabilities of correct detection, and of the various types of errors discussed in Subsection 4.2.1 earlier.

For example A_3^1, A_3^2 A_3^3 are subintervals where the measurement results of each test μ_1= A1C, μ_2= FPG, μ_3= OGTT can take values, given that the disease state is 3. The measurements from each test are real numbers. Figure 9 illustrates the subintervals where the measurement results from test μ_1= A1C take values, and the state estimate these subintervals imply. Their (disjoint) union forms the interval A^1. These measurements and subintervals are identical with the two threshold policies described earlier in Subsection 4.2.1.

Values of y_μ	A_1^1	A_2^1	A_3^1
Estimated state $\hat{x}$	1	2	3

Fig. 9: Subintervals in the test measurement space related to disease state estimation

As already mentioned, in this most practical scenario, we need to use the Controlled Hidden Markov Chain version of our model, as the states are not directly observable and cannot be used in the feedback for intervention selection. Rather the disease state is estimated based on the results of one or more diagnostic tests and then the intervention (or interventions) is decided and applied. We will develop the mathematical methodology and analytics for doing this partially observed casein future work; see [87] for some initial results.

Subsection 4.2.3: ***Interventions***

Interventions are selected and applied at each time instant with the purpose of improving the health of the patient. In our model we assume a discrete set of interventions including various medical procedures, medications, diet strategies, exercise strategies, technology aids. Thus $u(t) \in \{u_1, u_2, ..., u_{n_u}\}$. These interventions are selected on the basis of disease state estimates, projections on their effectiveness and benefit given the state and history of the patient as well as costs. The selection of interventions is referred to as ***strategies.*** A strategy involves a sequence of interventions applied at various times. The term strategy as used here is inspired by similar usage in engineering and especially in systems and control [60, 83]. Further, we consider in this study finite horizon problems. The finite time horizon could be the lifetime of the patient or the treatment horizon. In this study consider typically a treatment period of 10 years.

In all examples, simulations and numerical experiments we have included in this thesis we have considered four types of interventions, and ten specific interventions form these types. The four types of interventions considered are:

a) ***Technology*** $u_{(1,2)}$: we assume that one therapy uses communication wireless technology (telemedicine, EHR), while the other intervention is not using such technologies;

b) ***Diet*** $u_{(3,4,5)}$: we have three types of diets one that has as a target of losing 5% of weight, another with 10% and the last one with 15%.

c) ***Exercise*** $u_{(6,7,8)}$: we assume that a person can burn with 30 minutes of daily exercise 100, 200, 300 calories respectively.

d) ***Medication*** $u_{(9,10)}$: we have two interventions one that includes a light medication metformin and another one including strong insulin.

If the state is known, or, as is more appropriate for the most realistic models (realistic from the point of view of current medical practice), estimated on the basis of diagnostic tests results, medical practice standards and the reasoning and experience of medical practitioners result in a ***selection process for these interventions***; one can think of these selections as ranking the interventions in some order of preference. Figure 10 below illustrates one such set of preferences based on the disease state. More general conditioning including more detailed information about the patient can be incorporated in the same manner. Our overall MBSE healthcare support system and its operational context, as illustrated in Figure 1, supports the incorporation of additional information, models and statistics.

$\hat{x}$	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
1	0	1	0	0	0	1	0	0	0	0
2	1	0	0	1	0	0	1	0	1	0
3	1	0	0	0	1	0	0	1	0	1

Fig. 10: Illustrating a scenario with preferred interventions given the disease state

Although we have mostly considered a single intervention per time-step, our models and methods can easily incorporate combinations of "simultaneous" interventions by simply designating such combinations and their characteristics as additional possible interventions.

Chapter 5: **Evaluation Metrics**

Our dynamical model has the capability to produce sample paths, time histories of the progression of diabetes for a generic patient, in response to life-style, exercise or no exercise, other interventions and medications, and various administered diagnostic tests, form the ones described above. We are interested to develop metrics that will be used to evaluate these medical time histories and therefore indirectly the tests and interventions used in each case (in each time history). Thus to complete the Systems Engineering Model of Diabetes Mellitus, both at the disease as well as at the health care process levels, we next proceed to describe the metrics that we will use. Clearly one can add to the ones we describe below quite easily (see Chapter 6, pages 64 - 65, and Chapter 9 of this thesis).

One performance metric that we are interested in, is **cost**. The cost model we use is pretty straightforward. There is a cost for every test and every intervention used,

denoted by $C_u(u_k) = c_{u_k}$ and $C_\mu(\mu_l) = c_{\mu_l}$ respectively. The cost model we use is additive, that is the total cost for anyone time history for each patient is the sum of the cost of the tests, denoted by $C_\mu^{total}(i, m_i)$, and of the cost of the interventions, denoted by $C_u^{total}(i, m_i)$, used in the particular time history m_i. Our cost values include equipment, personnel and other costs. Clearly one could easily develop more detailed models by breaking down the components of each cost. Also, we can use costs data from doctors and hospitals directly in our model, or indirectly by using them to fit a functional model of each cost and its components (see Figure 1). Since our main goal is to develop and demonstrate a fundamental framework, we will not pursue these additional cost details in this thesis.

Thus for our work and analysis we have

$$C_\mu^{total}(i, m_i) = \sum_{t=1}^{N_{T,\Delta}} C_u(u(t)) \quad (9)$$

$$C_\mu^{total}(i, m_i) = \sum_{t=1}^{N_{T,\Delta}} C_\mu(\mu(t)) \quad (10)$$

and

$$C^{total}(i, m_i) = C_\mu^{total}(i, m_i) + C_u^{total}(i, m_i) \quad (11)$$

Where, in these sums, the tests and interventions used at each time step of this particular time history are considered.

For our numerical experiments the cost of the tests μ is given by the vector of test costs, which includes both equipment, facilities and labor costs:

$$\left[c_{\mu_1} \quad c_{\mu_2} \quad c_{\mu_3}\right] = \left[89 \quad 97 \quad 126\right]. \qquad (12)$$

Similarly the cost of interventions u is given by the vector of test costs, which includes both equipment, facilities and labor cost (dietician/nutritionist, physician, case manager and pharmacist)

$$\left[c_{u_1} \quad c_{u_2} \quad c_{u_3} \quad c_{u_4} \quad c_{u_5} \quad c_{u_6} \quad c_{u_7} \quad c_{u_8} \quad c_{u_9} \quad c_{u_{10}}\right] = \qquad (13)$$
$$= (320, 300, 160, 320, 430, 150, 300, 400, 350, 620).$$

Patients are of different types. Indeed behavioral characteristics of patients with respect to their healthcare are a very important albeit very difficult to model factor. In this thesis we will consider **three types of patients** with respect to the attention and systematic care that they apply to their health care and to following the recommendations resulting from tests and visits with doctors, as well as following orderly the prescribed interventions. It is well a known, and unfortunately a well-documented fact, that many patients do not follow recommendations and interventions rigorously (and some not at all). Our interest is to develop a somewhat realistic performance metric for each health care time history generated by our model.

We call the resulting metric ***Health Care Quality***, denoted by $J_{hc}(i, m_i)$. Here ***the first argument i is the index assigned to a specific patient, while the second argument m_i is the index assigned to a specific time history (the m^{th}) associated with patient i.*** Clearly the number of periods (in most of our work in this thesis this will mean the number of years) that a generic patient is in states 1, or 2, or 3, in a given time history, indicates very well the health of the patient. Thus, in the case for example where $N_{T,\Delta}$ is 10 (as in most examples and computations in this thesis) if a patient finds herself/himself in state 1 for 9 or 10 years, her/his health (with respect to diabetes) would be excellent, while if she/he is in state 3, 9 or 10 years, then her/his health condition (with respect to diabetes) would be very poor. Thus the number of years that a patient is in each state in a time history can be used as a meaningful metric for the quality of the condition of the patient's health.

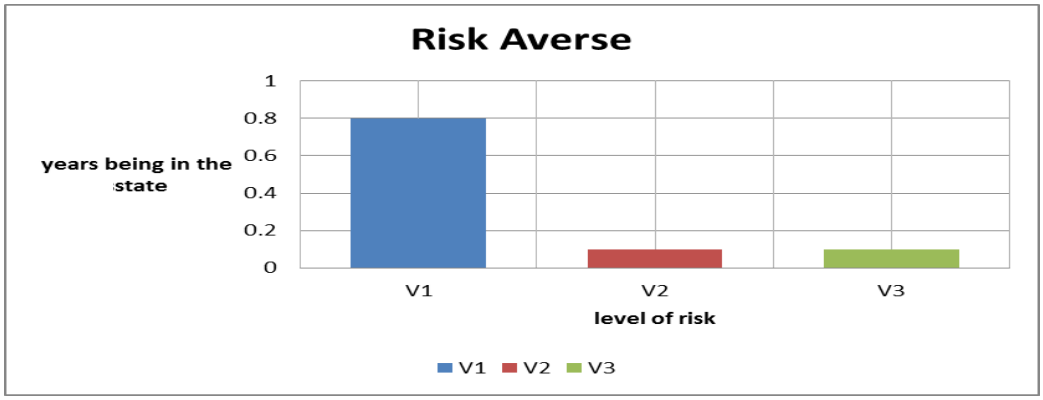
To capture the different behavioral types of patients we introduce weights representing the value (or significance) each patient places for being in each state of the model (recall Healthy, Pre-diabetic, Diabetic). We denote these weights by V_1^i, V_2^i, V_3^i , respectively, where the superscript *i* (we also use the superindex p_i to indicate patient *i* in various places and equations in this thesis) refers to the index of a specific patient, while the subscript refers to the states 1, 2, 3. These weights take real nonnegative values between 0 and 1, and their values sum to 1, to better indicate the relative value (or significance) that a

patient places in her/his health state with respect to diabetes. Thus we have the following:

$$\left.\begin{array}{l} V_1^i, V_2^i, V_3^i \in [0,1], \\ V_1^i + V_2^i + V_3^i = 1, \end{array}\right\} \forall i = 1, 2, \ldots, N_p \qquad (14)$$

where N_p is the number of patients in the study. Using these weights we can define three types of patients. The **"Risk Averse"** (with respect to the risk of getting sick with diabetes) patient may have representative weights as shown in Figure 11 below. That is for this type V_1^i is much larger than V_2^i , and V_3^i is almost zero. The **"Risk Indifferent"** patient may have representative weights as shown in Figure 12 below. That is for this type V_1^i is larger than V_2^i and V_3^i, but significantly smaller than the V_1^i for a risk averse patient. V_2^i and V_3^i have values that are almost the same. The **"Risk Taker"** patient type may representative weights as shown in Figure 13 below. That is for this type V_1^i and V_2^i are both larger than V_3^i, but taking almost the same value both. V_3^i takes values smaller than V_1^i and V_2^i but larger than the V_3^i for a risk averse patient, and close to the V_3^i for a risk indifferent patient.

As indicated earlier, the number of periods that each patient, in each generated time history, finds herself/himself in each of the three states is an important health condition



$$V_1^i = 0.8, V_2^i = 0.1, V_3^i = 0.1$$

Fig. 11: Risk averse patient type, where the Health Value weights have values

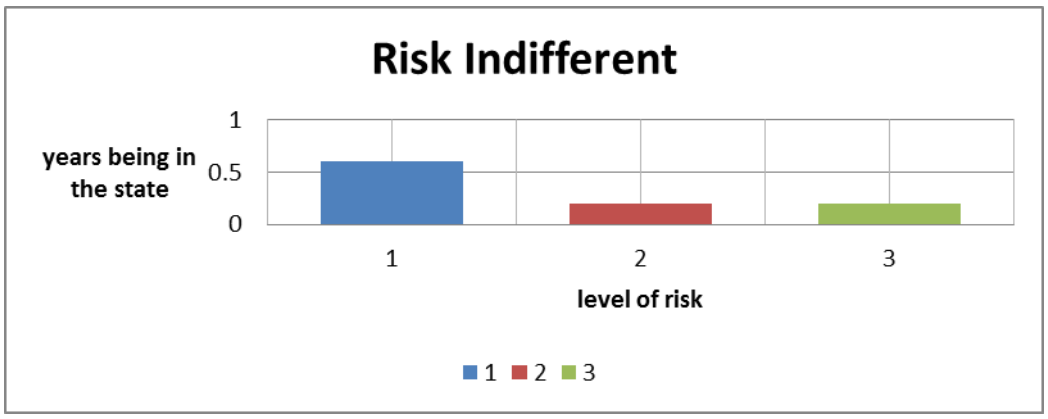


Fig. 12: Risk indifferent patient type, where the Health Value weights have values

$$V_1^i = 0.6, V_2^i = 0.2, V_3^i = 0.2$$

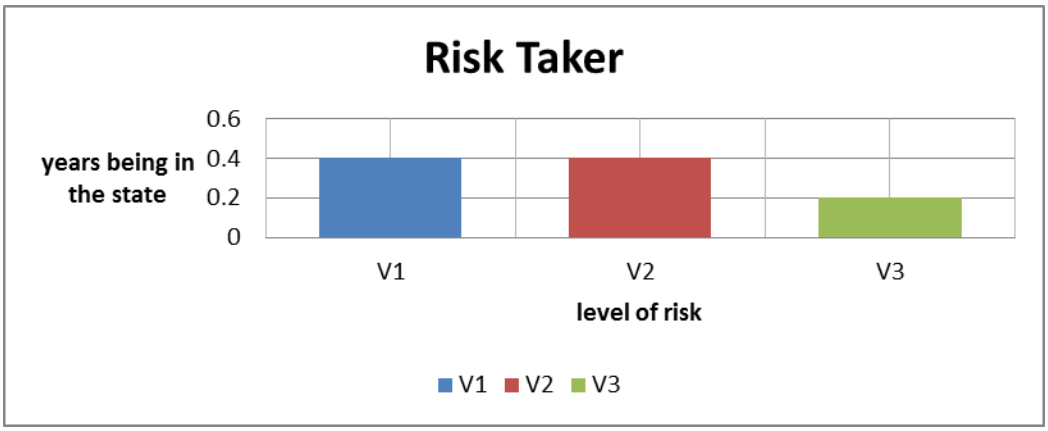


Fig. 13: Risk taker patient type, where the Health Value weights have values

$$V_1^i = 0.4, V_2^i = 0.4, V_3^i = 0.2$$

metric. As the state transitions depend explicitly on the tests and interventions applied at each time period, these numbers of periods in each state constitute a practical and useful health care value (or quality) metric. Thus we will also compute and include in our quantitative model and evaluations the three counting statistics for each patient and each time history from our model:

$$\begin{aligned} O_1^i(m_i) &= \text{number of periods, from } N_{T,\Delta} \text{ total, patient } i \text{ is at state 1 (i.e. is Healthy)} \\ O_2^i(m_i) &= \text{number of periods, from } N_{T,\Delta} \text{ total, patient } i \text{ is at state 2 (i.e. is Pre-diabetic)} \\ O_3^i(m_i) &= \text{number of periods, from } N_{T,\Delta} \text{ total, patient } i \text{ is at state 3 (i.e. is Diabetic)} \end{aligned} \quad (15)$$

Note that the fractions

$$F_1^i(m_i) = O_1^i(m_i) / N_{T,\Delta}, F_2^i(m_i) = O_2^i(m_i) / N_{T,\Delta}, F_3^i(m_i) = O_3^i(m_i) / N_{T,\Delta}, \quad (16)$$

may be given the interpretations of "probabilities" for patient *i* being in health state 1, 2, or 3, respectively.

Using the weights V_1^i, V_2^i, V_3^i and these counting statistics O_1^i, O_2^i, O_3^i, we can define several metrics for health care quality value (or value) for each patient and each time history, generated by our model. We will discuss how to construct and study several such metrics, and also how our MBSE health care management and support system can be used to evaluate new tests, therapies, and find the relative value of them, in a different section of the thesis. Let us first consider the following ***Health Care Quality metric***:

$$J_{hc}(i, m_i) = V_1^i * O_1^i(m_i) + V_2^i * O_2^i(m_i) + V_3^i * O_3^i(m_i) \quad (17)$$

Chapter 6: **First Method of Performance Evaluation and Tradeoff Analysis: Monte Carlo Simulation**

Section 6.1: Description of the EMCS Method

Using the metrics defined in Chapter 5 and the operation and the model developed in Chapter 4, we can now proceed to develop our first method for evaluating healthcare associated with each patient and each time history, and also aggregates of the same over subsets of patients, subsets of time histories or joint subsets of both. This first method uses the model in an exhaustive straightforward generation of all possible sample paths (time histories) for any number of patients, which we call Evaluation by Monte Carlo simulation (EMCS) (in our case it would be appropriate to call it Evaluation by Monte Carlo Markov Chain (MCMC) simulation) given the type of model we have developed and used. We mean more accurately that the model we use in this First Method is actually a Controlled Markov Chain and not a Hidden Markov Chain as we assume availability of the true state of the disease. MCMC has a rich history and there is extensive literature on all aspects of this method and uses of it [45-52]. This First Method is very accurate, as it considers all sample paths (time histories of the health and disease for each patient) but it is rather expensive computationally and may not scale to millions of patients and hundreds of tests and interventions easily. Nevertheless it provides a valuable benchmark against which to test more computationally efficient and scalable methods.

The EMCS method is described succinctly by the following steps (in algorithmic fashion):

Step 1: Run the model for the number of patients, horizon, time step, set of tests, set of interventions, and transition probabilities provided. Each time history starts for each patient at a randomly selected initial state, from the possible three available states. We denote by $x(0)$ the initial (randomly selected state). The integers 0, 1, 2, 3,…, 9, denote the time instants where decisions for tests and interventions are made. The t^{th} time period is then the interval $[t\text{-}1, t)$, for t = 1, 2, 3,…, 10. We do not make a decision for tests and or interventions at the final time 10, as there is no transition considered after that time. This is consistent with our operational model and experiments.

Step 2: Store the results of Step 1, as triples of arrays (vectors), one of dimension $N_{T,\Delta}+1$, two of dimension $N_{T,\Delta}$; a triple of arrays for each time history of each patient. The formats of these three arrays are as follows. The first contains the sequence of health states of this specific patient and the specific time history (we include both the initial state x (0) and the final state x (10) for each patient), the second the sequence of tests used at each time step and the third the sequence of interventions used at each time step. For example, as is

typical with all our simulation and experiments in this thesis, when $N_{T,\Delta}=10$,

$n_\mu = 3,\ n_u = 10$, we may have, with the obvious (by now) notation, the three arrays

$$
\begin{aligned}
&s(i,m_i) = \{1,1,2,3,3,2,2,3,2,2,2\} \quad \text{, the state array (sequence)} \\
&\mu(i,m_i) = \{\mu_2,\mu_3,\mu_1,\mu_1,\mu_2,\mu_3,\mu_2,\mu_1,\mu_3,\mu_1\} \quad \text{, the test array (sequence)} \\
&u(i,m_i) = \{u_3,u_4,u_8,u_7,u_9,u_2,u_2,u_5,u_5,u_8\} \quad \text{, the intervention array (sequence)}
\end{aligned} \tag{18}
$$

There are overall $N_p \times n_x^{N_{T,\Delta}+1} \times (n_\mu \times n_u)^{N_{T,\Delta}}$ such triples, as there are the same number of time histories, if we just generate a single time history for each patient. If on the other hand we generate N_S random time histories for each patient the cardinality of both sets (i.e. of time histories and of triples of arrays) becomes $N_p \times N_s \times n_x^{N_{T,\Delta}+1} \times (n_\mu \times n_u)^{N_{T,\Delta}}$.

Step 3: Using these arrays compute the costs $C_\mu^{total}(i,m_i)$, $C_u^{total}(i,m_i)$, $C^{total}(i,m_i)$, the counting statistics $O_1^i(m_i), O_2^i(m_i), O_3^i(m_i)$, and the healthcare value metric $J_{hc}(i,m_i)$. The cardinalities of these sets of counting statistics, values of each cost, or pairs of metrics, are exactly the same as the cardinalities of the sets described in Step 2.

Step 4: Plot for each patient and time history pair (i,m_i), the pair of values $(C^{total}(i,m_i), J_{hc}(i,m_i))$ in the positive quadrant of the plane (where the vertical axis (y-axis) corresponds to the total cost C^{total} and the horizontal axis (x-axis) to the Health Care Quality metric J_{hc}) and determine the Pareto points. Again the cardinality of the set of these 2D points is easily computable from the cardinalities of the sets described in Step 2.

Figure 14 below provides a graphical illustration of the method and the computations and complexity (size) involved.

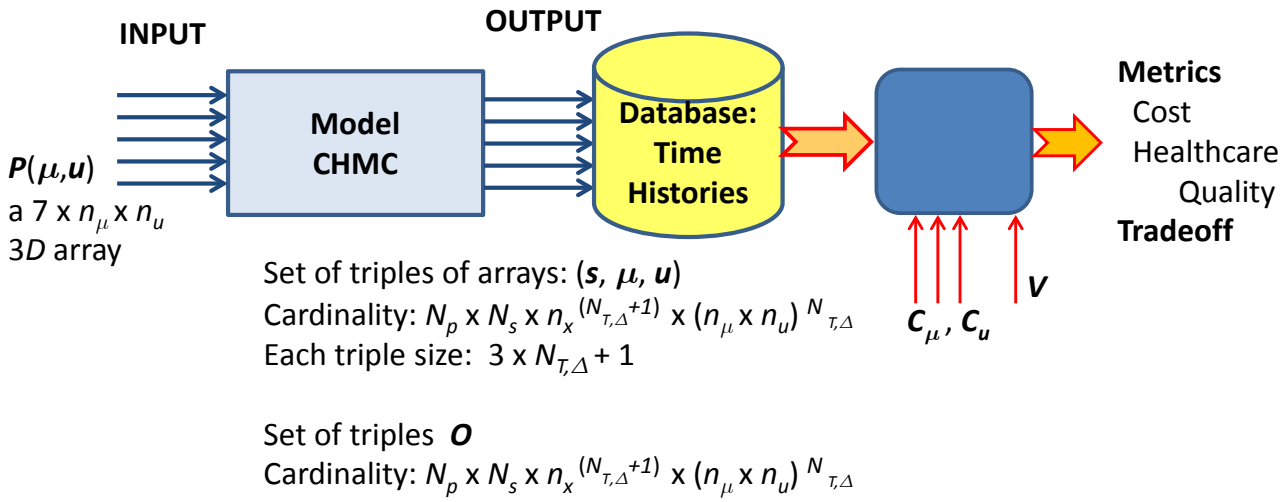


Fig. 14: Graphical illustration of the EMCS method of evaluation

Section 6.2: Input Parameters and Computational Details for EMCS

In this subsection we provide some details on the input parameters we used in our experiments and simulations with the EMCS Method, using the Disease model described in Section 4.1 of this thesis. We also describe the computations performed in some detail. Our entire MBSE system was built with MATLAB. We follow the description summarized in Figure 14 of Section 5.1 above.

The number of patients we have used is 10.000, the time period is 10 years, we have 3 tests that can be given to a patient (denoted by μ as already noted), we have 10 interventions that can be applied to a patient (denoted by u as already noted), and 3 disease states (denoted by x as already noted).

We set 30 matrices describing state transition probabilities, parametrized by the tests and interventions; that is the values of the elements $P_{ij}(\mu,u)$ were carefully selected based on literature reviews and simulation modeling reports [27-44, 55-59, 63-75, 81] to reflect the dependence of the transition probabilities on the selection of the tests and interventions used at each time step. Thus each matrix is a 3 by 3 table that shows for every combination of test and treatment what will be the effect in every state transition probability. In this simpler model, where we allow only one hop transitions in the input state transition probability matrices, the 3 by 3 input tables are tri-diagonal (i.e. the 31 and 13 elements are zero)) and have only 7 non-zero elements (7 parameters). In our model the elements $P_{ij}(\mu,u)$ do not depend explicitly on time. As already mentioned our models and methods can easily accommodate more complex models incorporating explicit dependence of the transition probabilities on time.

The second important input is the values for the costs for each test available and every intervention available. We refer to Section 3.2 and Subsection 4.2.3 for the descriptions of the available tests and interventions respectively. These values were determined as reasonable inputs based on literature review and reports from disease and treatment simulations [22-44, 55-59, 63-75, 81].

We constructed the required input arrays as follows.

C1 = [200, 100, 100, 200, 250, 100, 200, 300, 200, 500].

Each entry in this array represents the cost of each intervention from u_1 up to u_{10}. The order of entry is the same as in the description of the interventions provided in Section 4.2.3 of this thesis. That is the first cost is for u_1 and the last for u_{10} .

C2 = [120, 200, 60, 120, 180, 50, 100, 100 ,150 ,120]

Each entry in this array represents the labor cost that is related to the use of each intervention.

Finally we created the array Cu = C1 + C2, by component-wise addition of the elements of the two arrays. Each entry of the array Cu is the total cost of intervention cost and labor cost for implementing the intervention. These are the values (numbers that correspond to the costs $C_u(u_k)$ or c_{u_k} introduced at the beginning of Chapter 5.

Similarly for the costs of the tests we used the array

Cmu = [89, 97, 126] .

Each entry in this array represents the cost of each test and this cost includes the labor cost from the health care provider, the labor costs, the equipment and facility costs, supplies costs etc.. The order of entry is the same as in the description of the tests

provided in Section 3.2 of this thesis. That is the first cost is for μ_1 and the last for μ_3.

These are the values (numbers that correspond to the costs $C_\mu(\mu_l)$ or c_{μ_l} introduced at the beginning of Chapter 5.

We note that the values of these costs do not depend explicitly on time. This may arise in practical situations where the costs of health care delivery change due to labor cost changes, insurance policy changes, new treatments being discovered etc. Again, such time varying costs can be easily incorporated in our framework and analytical methods. Their impact on the EMCS method is just increasing computational complexity.

The third important input is input of variables used for the computation of the Health Quality metric. We only need to enter the values for the weights indicating how much (relatively) each type of patient (from the three we have considered) values the state of health (and disease) she/he is in. These are the weights described in Chapter 5 of this thesis and depicted in Figures 11, 12 and 13 in Chapetr 5, for the three types of patients we considered. W1 is the array of weights for a typical risk averse type patient; we assume that a risk averse patient will give value of 80% to remain in healthy state, 10% to pre-diabetic and 10% to diabetic. Thus

$$W1 = [0.8, 0.1, 0.1].$$

W2 is the array of weights for a typical risk indifferent patient; we assume that a risk indifferent patient will give value of 60% to remain in healthy state, 30% to pre-diabetic and 10% to diabetic. Thus

$$W2 = [0.6, 0.3, 0.1].$$

W3 is the array of weights for a typical risk taker patient; we assume that a risk taker patient will give value of 50% to remain in healthy state, 30% to pre-diabetic and 20% to diabetic. Thus

W3 = [0.5,0.3,0.2];

We note that the values of these weights do not depend explicitly on time. This may arise in practical situations where the value each patient gives to her/his health state may change, for example due to age (young people are more risk taker patients than older people), family state (married, pregnant, etc.), financial condition and security/insecurity, health insurance policies (e.g. recent policies rewarding health maintenance and well-being) and several other exogenous factors. Further, in reality these weights will vary from patient to patient, at least among groups of patients, even within the same category (i.e. the weights for two risk averse patients will be different). Again, such time varying weights and variation of weights among patients, can be easily incorporated in our framework and analytical methods. Their impact on the EMCS method is just increasing computational complexity. We would also like to emphasize again that, in the context and interconnected framework depicted in Figure 1, these variables can be "learned" from clinical data and databases and/or models and simulations of the disease progression and management.

One may want to call the three arrays W1, W2, W3, **Risk Profiles**, to emphasize that they can be interpreted from a risk perspective and theory, this fact actually inspired the naming of the patients categories as risk averse, risk indifferent and risk taker respectively.

As described in Chapter 5 and Section 6.1 above the Health Care Quality metric is computed using these value weights (risk profiles) and the counting statistics

(or as they are also called in the statistics and probability field **occupation statistics**)

$$F_1^i(m_i) = O_1^i(m_i) / N_{T,\Delta},\ F_2^i(m_i) = O_2^i(m_i) / N_{T,\Delta},\ F_3^i(m_i) = O_3^i(m_i) / N_{T,\Delta},$$

from equation (16) above. In the computations these histogram or history counts are represented by the array Rs, whose elements count the number of times the state of a patient has taken the value healthy or 1, pre-diabetic or 2 or diabetic or 3, all divided by the number of years (or time intervals in general) included in the horizon of the study; we have used 10 years. Thus

Rs = histc (X,[1,2,3]) / Ny;

where Ny is the variable for the number of years. As per equation (17) of Chapter 5 above, the value of the Health Care Quality metric is computed as the inner product of the weight arrays (i.e. the Wi's) and the counting statistics arrays (i.e. the Rs'). We have computed the values of this metric for each of the 10,000 patients and used them for tradeoff analysis. Thus the Health Care Quality metric for a risk averse, risk indifferent and risk taker patient is computed by the inner products

Perf1=W1*Rs ;

Perf2=W2*Rs ;

Perf3=W3*Rs .

One can use our disease model and MBSE health care management and support system to compute many other metrics. For example one can compute the number of state transitions in a given time history of a specific patient from a

worse health state to a better health state, namely number of transitions from state 3 to state 1, from state 2 to state 1 and from state 3 to state 2. For a specific patient p_i and time-history m_{p_i}, let us denote these counting transition statistics respectively as

$$\begin{aligned} Q_{3\to1}^{p_i}(m_{p_i}) &= \text{Number of transitions from state 3 to state 1 in time-history } m_{p_i} \\ Q_{3\to2}^{p_i}(m_{p_i}) &= \text{Number of transitions from state 3 to state 2 in time-history } m_{p_i} \\ Q_{2\to1}^{p_i}(m_{p_i}) &= \text{Number of transitions from state 2 to state 1 in time-history } m_{p_i}. \end{aligned} \quad (19)$$

Depending on the metric one wants to create the sum of these good transitions denoted by

$$Q_{good}^{p_i}(m_{p_i}) = Q_{3\to1}^{p_i}(m_{p_i}) + Q_{3\to2}^{p_i}(m_{p_i}) + Q_{2\to1}^{p_i}(m_{p_i}) \quad (20)$$

or the whole vector of number of transitions denoted by

$$Q_{tr}^{p_i}(m_{p_i}) = [Q_{1\to1}^{p_i}(m_{p_i}), Q_{1\to2}^{p_i}(m_{p_i}), Q_{1\to3}^{p_i}(m_{p_i}), Q_{2\to1}^{p_i}(m_{p_i}), Q_{2\to2}^{p_i}(m_{p_i}), Q_{2\to3}^{p_i}(m_{p_i}), Q_{3\to1}^{p_i}(m_{p_i}), Q_{3\to2}^{p_i}(m_{p_i}), Q_{3\to3}^{p_i}(m_{p_i})] \quad (21)$$

may also be of interest. Let us create then a ***reward metric*** (like an incentive) for patients with time histories displaying good health transitions. To this end we can create a reward (utility) vector for state transitions as follows

$$R = [R_{1\to1}, R_{1\to2}, R_{1\to3}, R_{2\to1}, R_{2\to2}, R_{2\to3}, R_{3\to1}, R_{3\to2}, R_{3\to3}] \ . \quad (22)$$

Then a new metric, which we will call **Reward** (for good behavior) can be easily created by taking the inner product of the reward vector of equation (22) with the vector of transitions, where we set to zero the components of the reward vector that do not correspond to health improvement. We have used in our simulations the example

$$R^{p_i}(m_{p_i}) = R_{2\to1}Q^{p_i}_{2\to1}(m_{p_i}) + R_{3\to1}Q^{p_i}_{3\to1}(m_{p_i}) + R_{3\to2}Q^{p_i}_{3\to2}(m_{p_i}) \ . \quad (23)$$

The intuition and motivation about this Reward metric is that a patient should perceive utility benefit (improvement of her health) when a therapy improve her health state (decreasing the index of the disease Markov state). To induce the correct and strong benefit we give extra weight to transitions from state 3 to ideally 1, and give zero weight to therapies that did not improve the state. A patient should use all this combination of therapies that increase his health utility (improvement of health) or moving to the lowest numbered states. For this to be ensured the values of the reward vector components should satisfy a relation like $R_{3\to1} > R_{3\to2} > R_{2\to1}$.

Similarly several reward and penalty (by penalizing negative health state transitions) metrics can be used depending on how aggressive the policy is intended to be. Such metrics could be used for example to reward risk averse patients with lower insurance premiums, patients with good transitions, and penalize patients with bad transitions, etc.

Regarding the computations involved in the EMCS method, we provide some details below.

At the start the software will set the computations for the population size we set.

(I = 1:Np).

Later the software will produce data based on the Monte Carlo simulations of the controlled Markov chain model and will generate so many data strings as many patients are in the population. We actually did not run the entire possible set of samples due to its size. For every data string that we will generate from Monte Carlo

data (i) = Monte Carlo(P,C,Ny) ,

we will have a cost variable C

Cost (i) = data (i).C .

For every data of a patient we will generate for every year we will have a state history for that patient

X (:,i) = data(i).x .

Pareout1 is the subroutine (algorithm) that computes the Pareto frontier for a discrete set of points in the plane, based on the Health Care Quality metric (array W1) for risk averse and Cost for every one of the 10.000 patients

pareout1= paretoGroup ([(1./Perf1)',Cost']) .

Pareout2 is the subroutine that computes the Pareto frontier, based on the Health Care Quality metric (array W2) for risk indifferent and Cost for every one of the 10.000 patients

pareout2 = paretoGroup ([(1./Perf2)',Cost']) .

Pareout3 is the subroutine that computes the Pareto frontier, based on the Health Quality metric (array W3) for risk taker and cost for every one of the 10.000 patients

pareout3 = paretoGroup ([(1./Perf3)',Cost']).

Section 6.3: Output from the MBSE Model and System

We have created a very versatile system, which can be used for various types of studies, tradeoffs and answering "what-if" type of questions, as we will demonstrate in later Chapters of this thesis (see Chapter 9).

A key output form our MBSE system, employing the ECMS method is the computation of Pareto points that describe succinctly the relative value of treatments and tests vs. the overall health care quality of a patients time history. The program produces graphs in the plane, showing all pairs of points corresponding to a patient's time history and a sequence of tests and treatments. The horizontal axis (*x*-axis) represents the values of the Health Care Quality metric (based on health value weights or risk profiles) for typical risk averse, risk indifferent and risk taker patients. The vertical axis (the *y*-axis) represents the values of the total Cost (tests and treatments) for each patient and patient time-history. On each graph the Pareto points are indicated with red color.

The function that makes the graph in our program is:

```
plot (Perf1, Cost,'.', Perf1(pareout1), Cost(pareout1),'r.');
```

The Pareto points can be used to find efficient treatments, to evaluate insurance policies and co-payments, to compare the relative value of tests and treatments, and many other important questions. We provided a description of the capabilities of the model using the ECMS method in later Sections. We provide

below three examples of the produced graphs, one each for each type of patient. In Appendix 1, we include a whole set of graphs for 32 runs of our model for 10,000 patients.

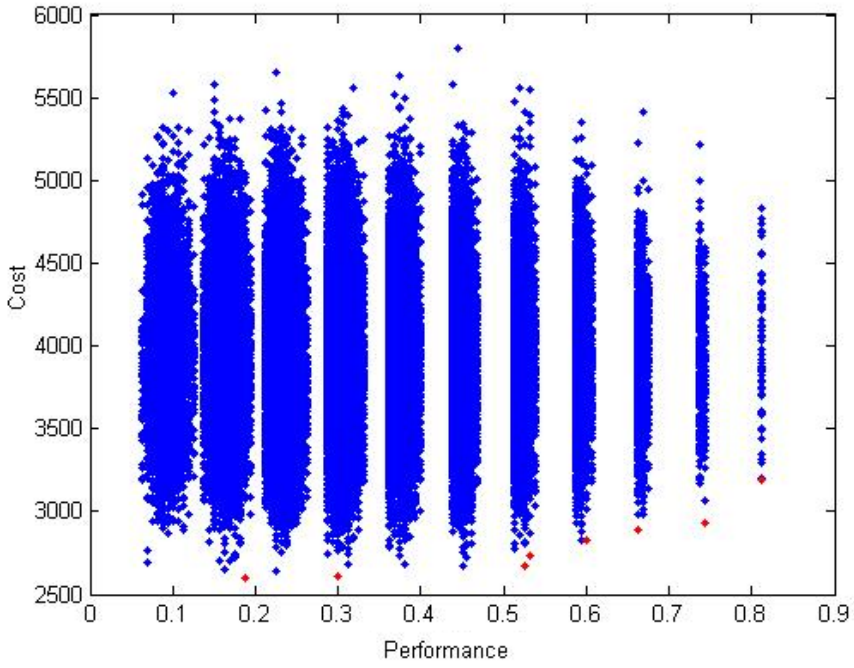


Fig. 15: Typical 2-D graph produced by our MBSE system that gives the Pareto frontier for a typical risk averse patient (Pareto points in red)

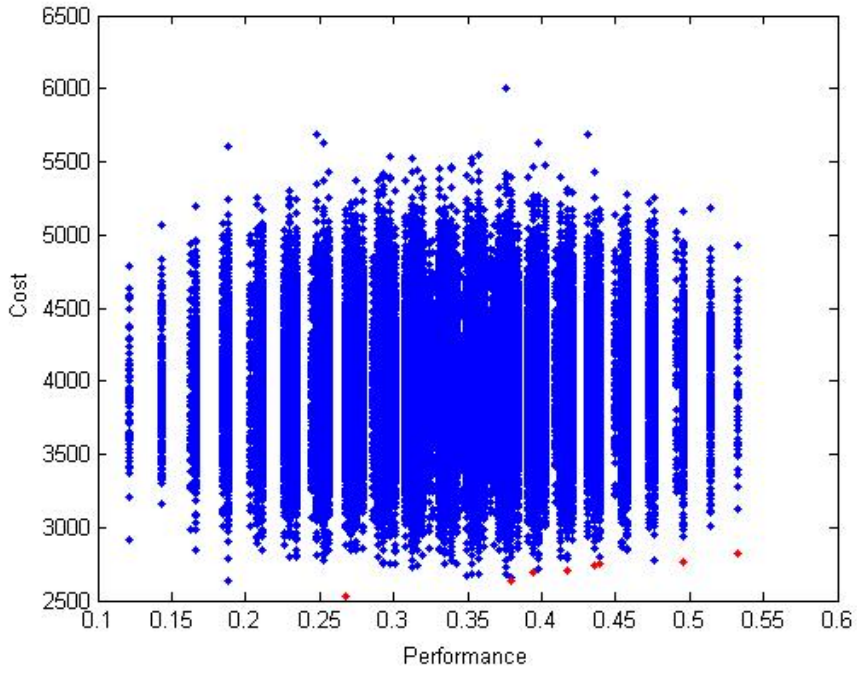


Fig. 16: Typical 2-D graph produced by our MBSE system that gives the Pareto frontier for a typical risk indifferent patient (Pareto points in red)

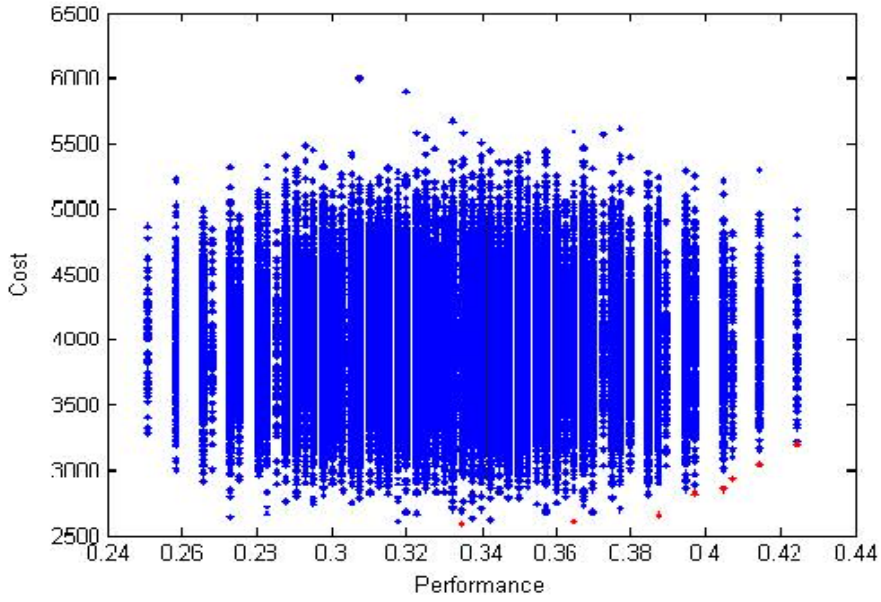


Fig. 17: Typical 2-D graph produced by our MBSE system that gives the Pareto frontier for a typical risk taker patient (Pareto points in red)

A quick comparison of these three graphs reveals some important, and expected, differences in the Pareto frontiers. For example the risk averse graph has much lower health quality values than the other two. The risk averse has higher cost values. These graphs are representative. We run numerous runs and in particular we run a set of 32 runs to have a proper random sample, in order to demonstrate the capabilities of the MBSE system we have developed. These capabilities are discussed in more detail in later Chapters of this thesis (see Chapter 9 in particular).

As discussed in Section 6.2 above the counting statistics (or occupation statistics) $O_1^i(m_i), O_2^i(m_i), O_3^i(m_i)$ when un-normalized, or $F_1^i(m_i), F_2^i(m_i), F_3^i(m_i)$ when normalized can be related to rewards and penalties and also can be used for answering several interesting questions (see Chapter 9 of this thesis). Furthermore, as discussed also in Section 6.2 above, the good and bad transition counts can be used to create reward and penalty metrics. For this reason we also produced three dimensional graphs form our MBSE system. In these 3D graphs three metrics are depicted: Cost, Performance (i.e. Health Care Quality metric), and Reward. Reward is computed for each patient and each time-history of a patient using equation (23) with a vector of rewards for transitions *R* (equation (22)) with values as follows

$$R = [0, 0, 0, 0.15, 0, 0, 0.31, 0.25, 0]$$

Three representatives of these 3-D graphs are shown below, one each for a typical risk averse, risk indifferent and risk taker patient. In Appendix 2, we include a whole set of 3-D graphs for 9 runs of our model but with 100,000 patients.

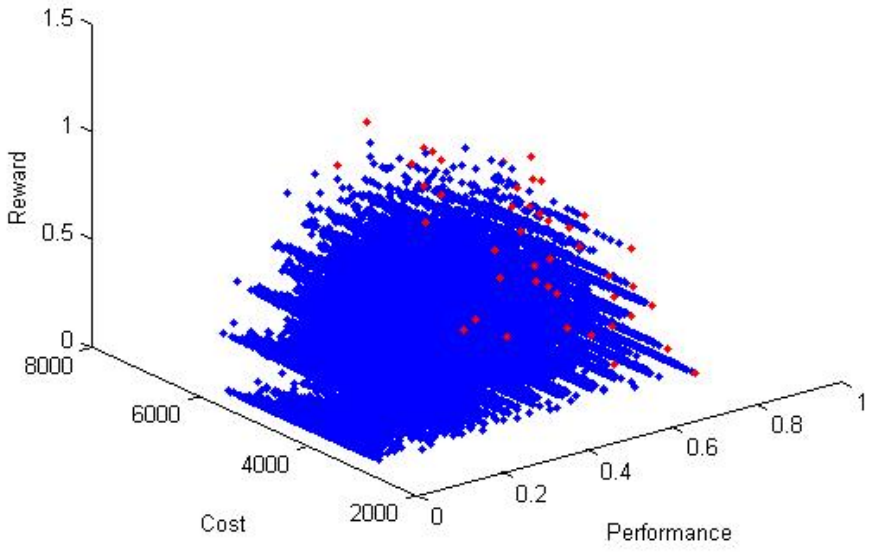


Fig. 18: Typical 3-D graph produced by our MBSE system that gives the Pareto frontier for a typical risk averse patient (Pareto points in red)

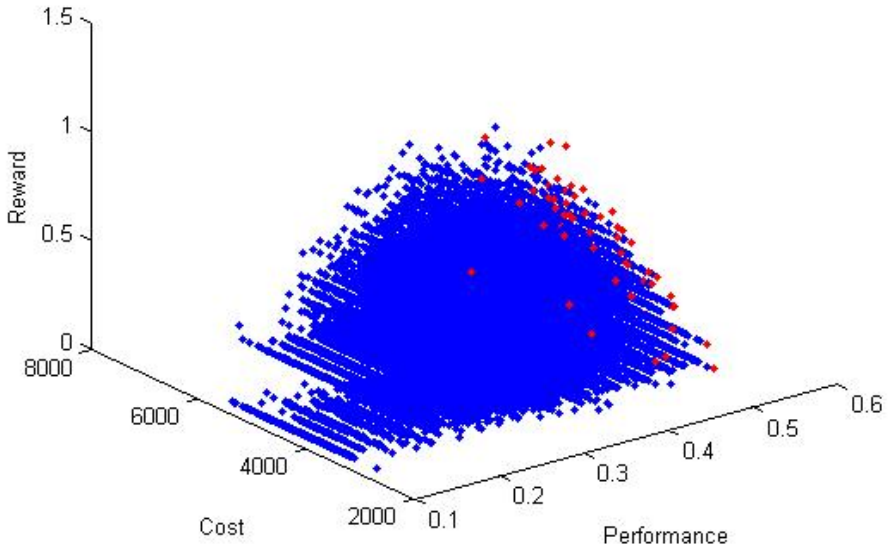


Fig. 19: Typical 3-D graph produced by our MBSE system that gives the Pareto frontier for a typical risk indifferent patient (Pareto points in red)

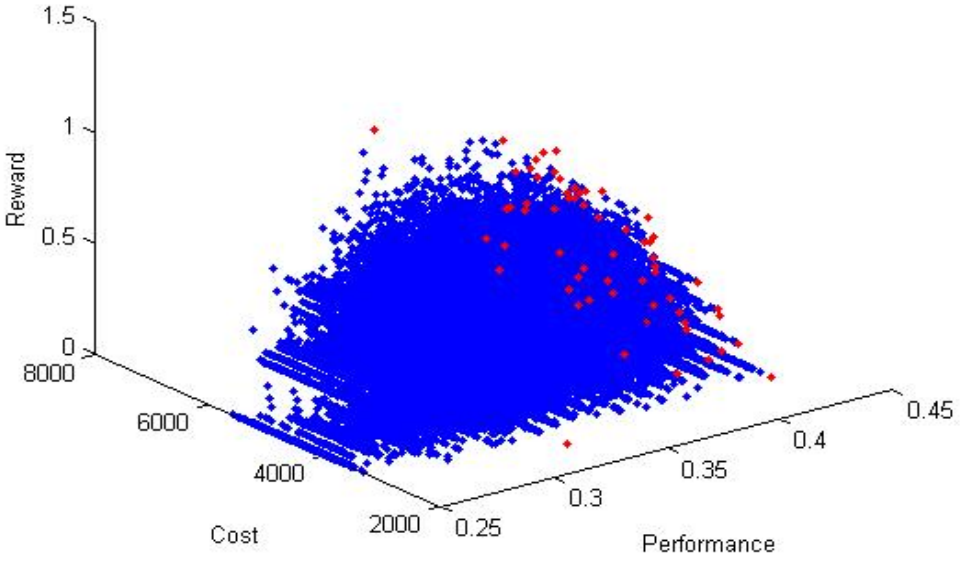


Fig. 20: Typical 3-D produced by our MBSE system that gives the Pareto frontier for a typical risk taker patient (Pareto points in red)

Finally, we close this Section with a discussion of the computational complexity and scalability of the EMCS method. This is important as we are interested in scalability to millions of patients and hundreds of interventions and tests. We have run many tests for 10,000 and 100,000 patients. We run all our tests on a laptop with the following characteristics:

$$\left.\begin{array}{l}\text{Intel core i5-4300ucpu}\\ \text{1.90-2.5 } Ghz\\ \text{8 Gb Ram}\\ \text{64bit operating system}\end{array}\right\} \quad (24)$$

Running EMCS with two metrics (and 2-D graphs) for 10,000 patients and 32 runs, took for the whole experiment 783sec. Running EMCS with three metrics (and 3-D graphs) for 100,000 patients took for the whole experiment 1,385 sec. So EMCS is definitely scalable and much better performance can be obtained with more powerful machines. Nevertheless as we are interested in superior efficiency, we will develop and analyze in the following sections of this thesis **alternative methods that decrease the execution time by two or more orders of magnitude**.

We have equipped our system and software implementation with several on-line diagnostics, that show computational performance at run time. Figure 21 shows screen dumps from these diagnostic and monitoring tools. The first two are from running EMCS with two metrics for 10,000 patients, while the third is from running EMCS with three metrics and 100,000 patients.

Profile Summary

Generated 20-Dec-2014 20:59:38 using cpu time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
main1_new	1	1384.996 s	110.346 s	
xlswrite	2548	867.597 s	8.982 s	
actxserver	2548	518.934 s	518.799 s	
monteCarlo	3000000	392.592 s	392.592 s	
xlswrite>ExecuteWrite	2548	337.844 s	81.990 s	
iofun\private\openExcelWorkbook	2548	174.645 s	166.385 s	
onCleanup>onCleanup.delete	3088	73.997 s	0.398 s	
xlswrite>@()xlsCleanup(Excel.file)	2548	73.579 s	0.145 s	
iofun\private\xlsCleanup	2548	73.434 s	73.131 s	

Profile Summary

Generated 20-Dec-2014 20:31:58 using cpu time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
main	1	2.009 s	0.573 s	
monteCarlo	10000	1.336 s	1.336 s	

Profile Summary

Generated 20-Dec-2014 22:20:23 using cpu time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
first_method_reward	1	783.910 s	4.289 s	
xlswrite	2080	728.682 s	2.718 s	
actxserver	2080	440.175 s	440.000 s	
xlswrite>ExecuteWrite	2080	284.417 s	69.104 s	
iofun\private\openExcelWorkbook	2080	145.377 s	137.871 s	

Fig. 21: Typical monitoring and on-line diagnostics from running EMCS. First two are form runs with two metrics and 10,000 patients, while the third is with three metrics and 100,000 patients.

Chapter 7: **Second Method of Performance Evaluation and Tradeoff Analysis: Fully Observable Multi-criteria Optimization**

Section 7.1: Description of the FOMCO Method

In this subsection we formulate a multi-criteria optimization approach to directly compute the Pareto points and associated (to these Pareto points) selection of tests and interventions. Multi-criteria optimization is a key methodology to perform tradeoff analysis and design space exploration in Systems Engineering [17-21]. The method we describe in this section uses feedback between the disease states and the selection of the tests and interventions to be applied at each time instance. More precisely the method automatically computes the tests and interventions to be applied at time t as functions of $x(t)$. Because it uses explicitly the state of the disease it is called **Fully Observable**, *vs* other methods, we described briefly in Subsection 4.2.2 pages 44-45 of this thesis, where the state is not available, and only estimates of the state based on the scores and results of diagnostic tests are available; this latter case is called **Partially Observable** [60, 83]. We will show that this second method, which we call **Fully Observable Multi-criteria Optimization (FOMCO)** saves tremendously in computational time in achieving similar tradeoff analysis results as the exhaustive simulation based first method (the EMCS Method described in Chapter 6). These computational savings are

important as we scale and apply our methods and tools to very large real-life problems and data sets involving millions of patients and numerous clinics and healthcare providers.

As described in the previous Chapters we have a dynamical stochastic system, describing the diabetes 2 disease progression, a CHMC model with n_x states. We denote as before the state of the disease model at time t by $x(t)$. In this second method we again select a single test and intervention at each time instant for the next time period, denoted by $\mu(t)$ and $u(t)$ respectively. Here however, in contrast with the simulation based performance analysis method (i.e. the EMCS method), we select the sequence of tests and interventions based on multi-criteria optimization. The efficiency achieved is precisely due to the fact that with this method we do not consider at all sequences of tests and interventions that lead to health care time histories that are dominated by others, from a precise tradeoff analysis point of view. In all our models the tests and interventions are selected from a finite set of options with cardinalities n_μ and n_u respectively. There are various ways that we can define the optimization problems underlying our optimization-based tradeoff analysis. One is to introduce the binary variables $X_l^\mu(t),\ l \in \{1,2,\ldots,n_\mu\},\ t = 0,1,2,\ldots,N_{T,\Delta}-1$, and $X_k^u(t),\ k \in \{1,2,\ldots,n_u\},\ t = 0,1,2,\ldots,N_{T,\Delta}-1$. This formalism is in-line with the so called component selection class of problems in Systems Engineering [17, 18].

That is:

$$X_l^{\mu}(t)=\begin{cases}1, \text{ if at time instant } t \text{ the test } \mu_l \text{ was performed}\\ 0, \quad \text{otherwise}\end{cases}$$
$$X_k^{u}(t)=\begin{cases}1, \text{ if at time instant } t \text{ the intervention } u_k \text{ was performed}\\ 0, \quad \text{otherwise}\end{cases} \tag{25}$$

Let us also introduce the binary variables $\Xi_j^x(t),\ j\in\{1,2,...,n_x\},\ t=0,1,2,...,N_{T,\Delta}$, to track the state of the system (in our case the diabetes 2 disease) through time. That is:

$$\Xi_j^x(t)=\begin{cases}1, \text{ if at time instant } t \text{ the state is } x_j\\ 0, \quad \text{otherwise}\end{cases} \tag{26}$$

Using these binary variables we can write the various metrics we have introduced as follows. For a patient i and time history m_i the cost of tests and interventions applied is

$$C^{total}(i,m_i)=C_{\mu}^{total}(i,m_i)+C_u^{total}(i,m_i)=\sum_{t=0}^{N_{T,\Delta}-1}C_{\mu}(\mu(t))+\sum_{t=0}^{N_{T,\Delta}-1}C_u(u(t))=$$
$$=\sum_{t=0}^{N_{T,\Delta}-1}\left(\sum_{l=1}^{n_{\mu}}C_{\mu}(X_l^{\mu}(t))+\sum_{k=1}^{n_u}C_u(X_k^{u}(t))\right). \tag{27}$$

In addition to ensure that only one test and one intervention are applied at each time instant we have to satisfy the constraints:

$$\sum_{l=1}^{n_{\mu}}C_{\mu}(X_l^{\mu}(t))=1, \text{ for all } t=0,1,2,...,N_{T,\Delta}-1$$
$$\sum_{k=1}^{n_u}C_u(X_k^{u}(t))=1, \text{ for all } t=0,1,2,...,N_{T,\Delta}-1 \tag{28}$$

Let the individual test and intervention costs be $c_1^{\mu}, c_2^{\mu}, ..., c_{n_\mu}^{\mu}$, $c_1^{u}, c_2^{u}, ..., c_{n_u}^{u}$. For the examples we employ in this thesis these vectors of costs are respectively $c_1^{\mu}, c_2^{\mu}, c_3^{\mu}$, $c_1^{u}, c_2^{u}, ..., c_{10}^{u}$. Then we can rewrite the total cost for a patient *i* and time history m_i as

$$C^{total}(i, m_i) = \sum_{t=0}^{N_{T,\Delta}-1} \left(\sum_{l=1}^{n_\mu} c_l^{\mu} X_l^{\mu}(t) + \sum_{k=1}^{n_u} c_k^{u} X_k^{u}(t) \right). \quad (29)$$

We note that this total cost for each patient *i* and time history m_i is deterministic. The reason, is that the costs for tests and interventions do not depend on the random state of the disease. Clearly one can develop more complex disease models where the tests and interventions depend on the disease states, as for example was described in earlier sections of this thesis. A second important observation is that due to the additive model used the value of the total cost is invariant to permutations between the tests and interventions used. In real-life and practical applications one may want to introduce **causality constraints** between the tests applied and the interventions employed. Such additional properties and constraints can be easily accommodated in the framework developed.

Thus $C^{total}(i, m_i)$ is a deterministic function of the discrete valued decision variables

$$\mu(t) = \mu_l,\ l \in \{1, 2, ..., n_\mu\},\ t = 0, 1, 2, ..., N_{T,\Delta} - 1,\ \ u(t) = u_k,\ k \in \{1, 2, ..., n_u\},\ t = 0, 1, 2, ..., N_{T,\Delta} - 1,$$

or equivalently of the binary valued decision variables

$X_l^{\mu}(t),\ l \in \{1,2,...,n_{\mu}\},\ t = 0,1,2,...,N_{T,\Delta} - 1,\ \ X_k^{u}(t),\ k \in \{1,2,...,n_u\},\ t = 0,1,2,...,N_{T,\Delta} - 1.$

To emphasize these dependencies, and for clarity in these dependencies, we write $C^{total}(i, m_i)$ as $C^{total}(i, \boldsymbol{\mu}, \boldsymbol{u})$, where $\boldsymbol{\mu} = [\mu(0), \mu(1), ..., \mu(N_{T,\Delta} - 1)]^{\mathrm{T}}$ and $\boldsymbol{u} = [u(0), u(1), ..., u(N_{T,\Delta} - 1)]^{\mathrm{T}}$. Equivalently we write $C^{total}(i, m_i)$ as $C^{total}(i, \boldsymbol{X}^{\mu}, \boldsymbol{X}^{u})$, emphasizing the binary valued vectors. We note that $C^{total}(i, \boldsymbol{X}^{\mu}, \boldsymbol{X}^{u})$ is linear in the variables over which we optimize, the binary vectors

$\boldsymbol{X}^{\mu} = [X_l^{\mu}(t),\ l \in \{1,2,...,n_{\mu}\},\ t = 0,1,...,N_{T,\Delta} - 1]^{\mathrm{T}},\ \boldsymbol{X}^{u} = [X_k^{u}(t),\ k \in \{1,2,...,n_u\},\ t = 0,1,...,N_{T,\Delta} - 1]^{\mathrm{T}}.$

We next express the Health Care Quality metric for a patient *i* and time history m_i, $J_{hc}(i, m_i)$, in terms of these same discrete valued and binary valued decision variables. Clearly the counting statistics for each patient *i* and time history m_i are given by the expressions

$$O_m^i(m_i) = \sum_{t=0}^{N_{T,\Delta}} \Xi_m^x(t), \ \text{for } m \in \{1,2,...,n_x\}. \quad (30)$$

Therefore

$$J_{hc}(i, m_i) = \sum_{m=1}^{n_x} V_m^i O_m^i(m_i) = \sum_{m=1}^{n_x} V_m^i \left(\sum_{t=0}^{N_{T,\Delta}} \Xi_m^x(t)\right) = \sum_{m=1}^{n_x} \sum_{t=0}^{N_{T,\Delta}} V_m^i \Xi_m^x(t). \quad (31)$$

Since the state sequence is a sequence of random variables, the value of this performance metric is a random variable. The value depends on the sequence of tests and interventions applied, as the transition probabilities, and therefore the actual transitions depend on the test and intervention applied at each time instant. Furthermore the value depends on the initial state of the disease for patient *i*, which we select randomly among the n_x states, with a uniform

probability mass function; i.e. $\Pr\{x(0) = x_m\} = 1/n_x,\ m = 1, 2, ..., n_x$. One can use various statistics of the random values of $J_{hc}(i, m_i)$ as the health care value metric, to be used in tradeoff analysis. In this thesis we used the average of the health care value metric for each patient *i*, over all possible time histories that can be generated by our CHMC model for the diabetes 2 disease, given the sequences of tests $\boldsymbol{\mu}$ and interventions $\boldsymbol{u}$. The average (or expectation) is taken over all possible transitions implied by the model, given a model of the randomness of the initial state, and the sequences $\boldsymbol{\mu}$ and $\boldsymbol{u}$. Then the expected health care value metric is given by

$$E[J_{hc}(i, m_i) \mid \boldsymbol{\mu}, \boldsymbol{u}] = \bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u}) = \sum_{m=1}^{n_x} \sum_{t=0}^{N_{T,\Delta}} V_m^i E[Z_m^x(t) \mid \boldsymbol{\mu}, \boldsymbol{u}] = \sum_{m=1}^{n_x} \sum_{t=0}^{N_{T,\Delta}} V_m^i \Pr\{Z_m^x(t) = 1 \mid \boldsymbol{\mu}, \boldsymbol{u}\}, \quad (32)$$

since $\Xi_m^x(t)$ is a binary valued (i.e. takes values 0 or 1) random variable. Obviously, $\Pr\{\Xi_m^x(t) = 1 \mid \boldsymbol{\mu}, \boldsymbol{u}\} = \Pr\{x(t) = m \mid \boldsymbol{\mu}, \boldsymbol{u}\}$, and we can **express this performance metric** as

$$E[J_{hc}(i, m_i)] = \bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u}) = \sum_{m=1}^{n_x} \sum_{t=0}^{N_{T,\Delta}} V_m^i \Pr\{x(t) = m \mid \boldsymbol{\mu}, \boldsymbol{u}\}. \quad (33)$$

Equivalently we can write the average Health Care Quality metric as a function of the binary vectors $\boldsymbol{X}^{\mu}$, $\boldsymbol{X}^{u}$, $\bar{J}_{hc}(i, \boldsymbol{X}^{\mu}, \boldsymbol{X}^{u})$, in a straightforward manner.

We are interested in analyzing the tradeoff between different pairs of sequences $(\boldsymbol{\mu}, \boldsymbol{u})$ from the perspective of the total Cost $C^{total}(i, \boldsymbol{\mu}, \boldsymbol{u})$ and average Health Care Quality $\bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u})$ metrics. This tradeoff analysis is important as it is necessary in order to find the "best" tradeoffs between the two conflicting objectives of

maximizing $\bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u})$ and minimizing $C^{total}(i, \boldsymbol{\mu}, \boldsymbol{u})$. These "best" tradeoff points are the **Pareto points** (also known as **non-dominated solutions**). Pareto efficiency, or Pareto optimality, is a state of allocation of resources in which it is impossible to make any one individual better off without making at least one individual worse off. The term is named after Vilfredo Pareto (1848–1923), an Italian economist who used the concept in his studies of economic efficiency and income distribution. The concept has applications in fields such as economics, engineering, and the life sciences [17-21]. Given an initial allocation of goods among a set of individuals, a change to a different allocation that makes at least one individual better off without making any other individual worse off is called a Pareto improvement. An allocation is defined as "Pareto efficient" or "Pareto optimal" when no further Pareto improvements can be made [17 - 21].

Pareto efficiency is a minimal notion of efficiency and does not necessarily result in a socially desirable distribution of resources: it makes no statement about equality, or the overall well-being of a society. The notion of Pareto efficiency can also be applied to the selection of alternatives in engineering and similar fields. It is a fundamental concept in tradeoff analysis as used in Systems Engineering. Each option is first assessed under multiple criteria and then a subset of options is identified with the property that no other option can categorically outperform any of its members. Given a set of choices and a way of valuing them, the ***Pareto frontier*** or ***Pareto set*** or ***Pareto front*** is the set of choices that are Pareto efficient [18, 19, 21]. By restricting attention to the set of

choices that are Pareto-efficient, a designer can make tradeoffs within this set, rather than considering the full range of every parameter. This key idea and construct essentially provides the savings in time between our First method (EMCS) and this Second method (FOMCO). These are the Pareto points we want to compute for the problem we are investigating in this thesis.

To compute these Pareto points for tradeoff analysis using the FOMCO method there are two principal techniques. The first and fastest technique combines the two metrics in a convex combination, resulting in the following single criterion optimization problem (often such methods are called **scalarization methods** [21, 82]):

(FOMCO-S) Find the pair of discrete valued vectors ($N_{T,\Delta}$ dimensional), $(\boldsymbol{\mu}^*, \boldsymbol{u}^*)$ that solve the optimization problem:

$$\max_{\boldsymbol{\mu},\boldsymbol{u}} [\lambda \bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u}) + (1-\lambda)(-C^{total}(i, \boldsymbol{\mu}, \boldsymbol{u})] \quad (34)$$

where $\lambda \in (0,1)$. The optimization is over the allowed finite sets of choices for tests and interventions, with cardinalities n_μ, n_u respectively. The pair $(\boldsymbol{\mu}^*, \boldsymbol{u}^*)$ is a Pareto point.

In formulating this so called "scalarized" metric we used the fact that minimizing a function is equivalent to maximizing its negative. This method computes Pareto points of the convexification of the Pareto frontier [18, 21] (i.e. the smallest convex set that includes the Pareto frontier). We should solve this

problem for various values of λ. In our computations we used $\lambda = 0.1, 0.2, 0.3, \ldots, 0.8, 0.9$, as well as much finer quantization of λ. Once the optimal vectors of tests and interventions have been found, one runs the CHMC disease model and obtains consistent (with respect to this selection of tests and interventions) state trajectories.

When using these scalarization methods a common problem appears when the two metrics can take significantly different values, the so called dynamic range problem [18, 19]; as is the case for our intended application, since the total cost $C^{total}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ can take values in several thousand while the Health Care Quality metric $\overline{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ takes values between 0 and 1. When the values of the one metric are so much larger than the values of the other, when they are combined in a convex combination like in equation (34), the values of the metric with the small values do not affect the tradeoff and thus the results can be quite erroneous. The correction to this well-known problem is to normalize the two metrics so that the both take values that can be compared. By far the best normalization method is to normalize these metrics by the difference between their maximum and minimum values over the possible decision variables. To this end we need to compute upper and lower bounds for the values of our two metrics $C^{total}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ and $\overline{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$.

To compute an upper bound for the Health Care Quality metric $\overline{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$, observe that it is computed according to the formula (see equation (33)):

$$\overline{J}_{hc}(p_i,\boldsymbol{\mu},\boldsymbol{u})=\sum_{t=0}^{9}\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m+\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(10)_m \quad (35)$$

But in our case, for each patient p_i , which can be of either risk averse, or risk indifferent or risk taker type, there are three weights $\{V_1^{p_i},V_2^{p_i},V_3^{p_i}\}$ and they do not differ for patients of the same type. Let

$$V_{\max}^r=\max\{V_1^r,V_2^r,V_3^r\},\quad V_{\min}^r=\min\{V_1^r,V_2^r,V_3^r\},\quad \text{for } r=1,2,3, \quad (36)$$

where we denote the type risk averse by *r* = 1, the type risk indifferent by *r* = 2, and the type risk taker by *r* = 3. Since in our computations we do not mix the patient types, we will use different normalizations for the Health Care Quality metric for each patient class. We designate the corresponding bounds by a superindex *r*. Thus using (35) and (36) we have

$$\begin{aligned}&\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m\le V_{\max}^r\sum_{m=1}^{3}\pi_{(\boldsymbol{\mu}^{\lambda,*},\boldsymbol{u}^{\lambda,*})}(t)_m=V_{\max}^r,\quad \text{for}\quad r=1,2,3\\&\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m\ge V_{\min}^r\sum_{m=1}^{3}\pi_{(\boldsymbol{\mu}^{\lambda,*},\boldsymbol{u}^{\lambda,*})}(t)_m=V_{\min}^r,\quad \text{for}\quad r=1,2,3.\end{aligned} \quad (37)$$

Using (37) in (35) we have

$$\begin{aligned}&\sum_{t=0}^{9}V_{\max}^r+V_{\min}^r=11V_{\min}^r\le\\&\le\overline{J}_{hc}(p_i,\boldsymbol{\mu},\boldsymbol{u})=\sum_{t=0}^{9}\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m+\sum_{m=1}^{3}V_m^{p_i}\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(10)_m\le\\&\le\sum_{t=0}^{9}V_{\max}^r+V_{\max}^r=11V_{\max}^r,\quad \text{for}\quad r=1,2,3.\end{aligned} \quad (38)$$

This leads to the consideration of the normalized Health Care Quality metric

$$\overline{J}_{hc}^{r,n}(p_i,\boldsymbol{\mu},\boldsymbol{u})=\overline{J}_{hc}(p_i,\boldsymbol{\mu},\boldsymbol{u})/(11(V_{\max}^r-V_{\min}^r))=\overline{J}_{hc}(p_i,\boldsymbol{\mu},\boldsymbol{u})/J_n^r,\ \text{for}\ r=1,2,3. \quad (39)$$

Next we compute similar bounds for the total cost

$$\sum_{t=0}^{9}\min\{c_1^{\mu}, c_2^{\mu}, c_3^{\mu}\}+\sum_{t=0}^{9}\min\{c_1^{u}, c_2^{u},...,c_{10}^{u}\}=10\,(c_{\min}^{\mu}+c_{\min}^{u})=C_{\min}^{total}\le$$

$$\le C^{total}(p_i,\boldsymbol{\mu},\boldsymbol{u})=\sum_{t=0}^{9}(C_{\mu}(\mu(t))+C_{u}(u(t)))\le \qquad (40)$$

$$\le\sum_{t=0}^{9}\max\{c_1^{\mu}, c_2^{\mu}, c_3^{\mu}\}+\sum_{t=0}^{9}\max\{c_1^{u}, c_2^{u},...,c_{10}^{u}\}=10\,(c_{\max}^{\mu}+c_{\max}^{u})=C_{\max}^{total}$$

This leads to the consideration of the normalized total Cost metric

$$C^{total,n}(p_i,\boldsymbol{\mu},\boldsymbol{u})=C^{total}(p_i,\boldsymbol{\mu},\boldsymbol{u})/(10\,((c_{\max}^{\mu}-c_{\min}^{\mu})+(c_{\max}^{u}-c_{\min}^{u})))=C^{total}(p_i,\boldsymbol{\mu},\boldsymbol{u})/C_n. \quad (41)$$

Clearly maximizing $\overline{J}_{hc}(p_i,\boldsymbol{\mu},\boldsymbol{u})$ is completely equivalent to maximizing $\overline{J}_{hc}^{r,n}(p_i,\boldsymbol{\mu},\boldsymbol{u})$, while minimizing $C^{total}(p_i,\boldsymbol{\mu},\boldsymbol{u})$ is completely equivalent to minimizing $C^{total,n}(p_i,\boldsymbol{\mu},\boldsymbol{u})$. Thus we will use the properly normalized values of the two metrics in our implementation of FOMCO-S, which we designate as FOMCO-SN (N for normalized). For clarity we state the FOMCO-SN problem below.

(FOMCO-SN) Find the pair of discrete valued vectors ($N_{T,\Delta}$ dimensional), $(\boldsymbol{\mu}^*,\boldsymbol{u}^*)$ that

solve the optimization problem:

$$\max_{\boldsymbol{\mu},\boldsymbol{u}}[\lambda\overline{J}_{hc}^{r,n}(i,\boldsymbol{\mu},\boldsymbol{u})+(1-\lambda)(-C^{total,n}(i,\boldsymbol{\mu},\boldsymbol{u})] \quad (42)$$

where $\lambda \in (0,1)$. The optimization is over the allowed finite sets of choices for tests and interventions, with cardinalities n_μ, n_u respectively. The pair $(\boldsymbol{\mu}^*, \boldsymbol{u}^*)$ is a Pareto point.

In the second technique for FOMCO tradeoff analysis, we compute the Pareto points by solving several optimization problems where one of the metrics is selected for optimization and all the others are used as constraints; the so-called ***ε-method*** [18, 19, 20, 21]. Thus we solve the following set of optimization problems.

(FOMCO-E) Find the pair of discrete valued vectors ($N_{T,\Delta}$ dimensional), $(\boldsymbol{\mu}^*, \boldsymbol{u}^*)$ that solve the optimization problems:

$$\begin{aligned} &\max_{\boldsymbol{\mu},\boldsymbol{u}} \ \bar{J}_{hc}(i, \boldsymbol{\mu}, \boldsymbol{u}) \\ &\text{subject to } C^{total}(i, \boldsymbol{\mu}, \boldsymbol{u}) \le \varepsilon \end{aligned} \tag{43}$$

where ε is varied over a set of appropriate values. This method computes exactly all the Pareto points, but it is more costly computationally than FOMCO-S.

Both problems can be solved by Dynamic Programming (DP) based methods [60 - 62, 83], applied to the appropriate metric.

Section 7.2: Description of Deterministic and Stochastic DP Algorithms

In this section, for completeness and ease of reference, we briefly review Dynamic Programming Algorithms and establish the associated nomenclature and notation. Dynamic Programming (DP) was introduced by Richard Bellman in 1954 [62] for recursively solving dynamic optimization, optimal control and multi-stage decision problems. It has been used extensively in many areas ranging from signal processing to bioinformatics. In this brief review we describe DP as close as possible to the problems we are investigating, and for the solution of which we will use the DP algorithm. The general problem that fits our interests is:

(GDP)

$$\begin{aligned} &\max_{\{v(t)\}} J(x(0), x(1), ..., x(T); v(0), v(1), ..., v(T-1)) \\ &\text{subject to} \quad i) \quad G(x(0), x(1), ..., x(T); v(0), v(1), ..., v(T-1)) \geq 0 \\ &\qquad\qquad ii) \quad v(t) \in \Omega(t) \text{ for each } t = 0,1,...,T-1 \\ &\qquad\qquad iii) \quad x(0) = x_0 \text{ given} \\ &\qquad\qquad \text{iv)} \quad x(T) \geq 0\,, \end{aligned} \tag{44}$$

where:

$x(t)$ is a vector of state variables that describe the state of the system at any point in time.

$v(t)$ is a vector of control variables which ***can be chosen*** in every period by the decision-maker.

$J(.)$ is the objective function (i.e. the metric we are optimizing) which is, in general, a function of all the state and control variables for each time period.

$G(.)$ is a system of inter-temporal constraints connecting the state and control variables.

$\Omega(t)$ is the set of allowed values for the control variables at time *t*.

Dynamic Programming works for problems that satisfy the following structural constraint: Both $J(.)$ and $G(.)$ are ***time separable***. That is:

$$J(x(0), x(1), \ldots, x(T); v(0), v(1), \ldots, v(T-1)) = U(0, x(0), v(0)) + U(1, x(1), v(1)) + \ldots + U(T-1, x(T-1), v(T-1)) + S(T, x(T)) \quad (45)$$

where $S(x(T))$ is a value function at the end of the decision process, where no further decisions are made. Furthermore, the $G(.)$ functions satisfy the ***Markov structural constraint***:

$$\begin{aligned} x(1) &= G(0, x(0), v(0)) \\ x(2) &= G(1, x(1), v(1)) \\ &\vdots \\ x(T) &= G(T-1, x(T-1), v(T-1)), \end{aligned} \quad (46)$$

where these are the ***state transition equations***, or the equations describing the ***evolution of the system state*** in response to the chosen controls. These structural constraints are essential for Dynamic Programming and they are the reason for the efficient recursions involved in DP.

Then the optimization or decision problem becomes:

(DPS)

$$\max_{\substack{v(t);t=0,1,\ldots,T-1 \\ v(t)\in\Omega(t)}} J(x_0, \boldsymbol{g}) = \sum_{t=0}^{T-1} U(t, x(t), v(t)) + S(T, x(T))$$

$$\text{subject to} \quad i) \quad x(t+1) = G(t, x(t), v(t)), \quad t = 0,1,\ldots,T-1, \quad (47)$$

$$ii) \quad x(0) = x_0, \quad \text{given},$$

where wed denote by $\boldsymbol{g}$ the control policy (or control law) $\{v(0), v(1), \ldots, v(T-1)\}$.

Bellman's method (Dynamic Programming) solves this problem by considering the larger class of ***sub-problems of this problem*** [60], corresponding to the same problem but starting at some time $t_0 > 0$. Namely, consider the problems below for each $t_0 = 0,1,\ldots,T-1$.

(DPSsub)

For each $t_0 = 0,1,\ldots,T-1$

$$\max_{\substack{v(t);t=t_0,1,\ldots,T-1 \\ v(t)\in\Omega(t)}} J(x_{t_0}, t_0, \boldsymbol{g}) = \sum_{t=t_0}^{T-1} U(t, x(t), v(t)) + S(T, x(T)) \quad (48)$$

$$\text{subject to} \quad i) \quad x(t+1) = G(t, x(t), v(t)), \quad t = t_0,\ldots,T-1,$$

$$ii) \quad x(t_0) = x_{t_0}, \quad \text{given}.$$

Let the solution (i.e. the optimal value of the objective or metric) of Problem (DPSsub) be defined as the ***value function*** $V(x_{t_0}, T-t_0)$. Then Bellman's ***Principle of Optimality*** [60-62, 83] asserts that any solution of Problem (DPS) on t = 0,...,T, which yields $x(t_0) = x_{t_0}$ must also solve Problem (DPSsub) on the range $t = t_0,\ldots,T$. Dynamic Programming follows from this principle and it essentially solves the larger Problem (DPS) by solving the smaller Problem (DPSsub) sequentially (recursively). Further, since t_0 is arbitrary, we can choose to solve the Problem (DPSsub) first for $t_0 = T$-1 first and then work backwards in

time. This gives the DP algorithm, the recursive step, for the period *T-k*, expressed below as the well-known ***Bellman Equation***:

$$\begin{aligned}
&\text{For} \quad k = 1, 2, \ldots, T \\
&V(x(T), 0) = S(T, x(T)) \\
&V(x(T-k), k) = \max_{v(T-k) \in \Omega(T-k)} \{U(T-k, x(T-k), v(T-k)) + V(x(T-k+1), k-1)\} \quad (49) \\
&\text{subject to:} \quad i) \quad x(T-k+1) = G(T-k, x(T-k), v(T-k)) \\
&\qquad\qquad\quad ii) \quad x(T-k) = x_{T-k}, \quad \text{given.}
\end{aligned}$$

This fundamental iteration can be also written as (to clarify the single step optimization involved):

$$\begin{aligned}
&\text{For} \quad k = 1, 2, \ldots, T \\
&V(x(T), 0) = S(T, x(T)) \qquad (50) \\
&V(x(T-k), k) = \max_{v(T-k) \in \Omega(T-k)} \{U(T-k, x_{T-k}, v(T-k)) + V(G(T-k, x_{T-k}, v(T-k)), k-1)\}
\end{aligned}$$

Solution of this one step maximization problem, given the form of the value function from the previous step, will yield a control rule:

$$v(T-k) = g(T-k, x(T-k)), \;\; k = 1, 2, \ldots, T, \quad (51)$$

that is a control law in ***state feedback form***. After going through the successive rounds of single period maximization problems, eventually we reach the problem at time zero, which is the original problem:

$$\begin{aligned}
&V(x(0), T) = \max_{v(0) \in \Omega(0)} \{U(0, x(0), v(0)) + V(x(1), T-1)\} \\
&\text{subject to:} \quad i) \quad x(1) = G(0, x(0), v(0)) \qquad (52) \\
&\qquad\qquad\quad ii) \quad x(0) = x_0, \quad \text{given,}
\end{aligned}$$

and which will also yield the control rule:

$$v(0) = g(0, x_0). \quad (53)$$

That is $V(x_0, T) = J(x_0, \boldsymbol{g}^*)$, the optimal value of the metric; $\boldsymbol{g}^*$ denotes the optimal policy. Knowing $x(0)$ and $v(0)$ we can use the state transition equations

to compute $x(1)$. Then knowing $x(1)$ and $v(1)$ we can compute $x(2)$ and so on, until all the values of the state $x(t)$ (the state time history), and controls $v(t)$ (the controls time history) are computed. The time history of the controls is called ***control policy*** and we will denote control policies generically by $\boldsymbol{g}$. Thus, we have obtained the complete solution of the optimization problem of interest, in an efficient manner (as compared with the exhaustive evaluation of all options – like in our EMCS First Method) due to the recursive computation involved, and the state feedback selection of tests and interventions.

The DP approach is very powerful, especially for the problems we have encountered in this thesis, and interested in. Namely dynamic optimization problems with discrete time, finite horizon, finite number of system states, and finite number of choices for controls or decisions. It allows all types of constraints on the state and controls, including control constraints that depend on the state, to be easily incorporated in the formulation and handled in the same manner. Furthermore, the formulation we described covers time varying problems, including time varying constraints, which are of interest for medicine and health care, as they can represent changing exogenous conditions of the patient, the treatment and the environment.

In our Second Method of performance evaluation and tradeoff analysis, the FOMCO method, we assume that we observe the state of the disease (i.e. the state of the disease is known to the health care professional at the decision instants with no error), and that the tests and interventions applied at each

decision time only depend on the state of the disease at that time. As the underlying model of the disease is stochastic, we need a small modification of the DP formalism just described. The state transitions are now stochastic and they are determined by a controlled Markov chain (CMC); the state is observed (known) at each decision time instant; the same type of model we developed for the diabetes II disease progression in Section 4 of this thesis. Let the transition probabilities of the underlying CMC be given by

$$P_{ij}(t,v) = \Pr\{x(t+1) = j \mid x(t) = i, v(t) = v\}, \quad (54)$$

and this denotes the probability that the state will transition to state *j* at the next time $t + 1$, when it is at state *i* at the present time *t* and control with value $v(t) = v$ is applied at time *t*. ***Markov policies*** are consistent with the results form DP; namely that DP when the state is observed provides optimal controls that are functions of the current state (feedback form). That is $v(t) = g(t, x(t))$. With a Markov policy $\boldsymbol{g}$ the dynamics of the controlled system are indeed governed by a Markov chain with state transition probabilities

$$P_{ij}(t,\boldsymbol{g}) = \Pr\{x(t+1) = j \mid x(t) = i, v(t) = g(t,i)\} = P_{ij}(t, g(t,i)), \quad (55)$$

That is precisely the CMC model class we used in Chapter 4. We denote generically by $\boldsymbol{g}$ such policies. So everything in our formulation is consistent. Then the stochastic version of the optimization problem (DPS) (equation (39)) is:

(StochDPS)

$$\max_{\substack{v(t);t=0,1,\ldots,T-1 \\ v(t)\in\Omega(t)}} J(x_0,\boldsymbol{g}) = E\{\sum_{t=0}^{T-1} U(t,x(t),v(t)) + S(T,x(T))\}$$

$$\text{subject to} \quad i) \quad x(t+1) = G(t,x(t),v(t),\varepsilon(t+1)), \quad t = 0,1,\ldots,T-1, \qquad (56)$$

$$ii) \quad x(0) = x_0, \quad \text{given},$$

where $\boldsymbol{g}$ denotes the Markov policy $\{v(t) = g(t,x(t)); t = 0,1,\ldots,T-1\}$, and E denotes expectation with respect to the joint multidimensional probability mass functions (pmf) of $\{x(t); t = 0,1,2,\ldots,T\}$ and $[v(t); t = 0,1,\ldots,T-1\}$. In fact, given $G(.,.,.,.)$ the only stochastic inputs to the system dynamics are the given initial condition x_0 (a random variable) and the finite sequence of random variables $\{\varepsilon(t); t = 1,2\ldots,T\}$. Here $\{\varepsilon(t); t = 1,2,\ldots,T\}$ is a sequence of independent (over time) random variables (e.g. like white noise). Since for Markov policies $\{v(t) = g(t,x(t)); t = 0,1,\ldots,T-1\}$, given $G(.,.,.,.)$ and the policy $\boldsymbol{g}$ (i.e. the functions $\{g(t,.); t = 0,1,\ldots,T-1\}$) the joint multidimensional probability mass functions of $\{x(t); t = 0,1,2,\ldots,T\}$ and $\{v(t); t = 0,1,\ldots,T-1\}$ are determined by the statistics of x_0 and $\{\varepsilon(t); t = 1,2,\ldots,T\}$. A very general, but basic, result in finite state Markov chain theory provides the complete equivalence of the sample path dynamical model above (through $G(.,.,.,.)$) with the more traditional one described through the state transition probabilities [45-51, 60 - 62, 83]. Therefore equivalently E denotes expectation with respect to the exogenous randomness at time 0, $I_0 = (\{x_0\}, \{\varepsilon(t); t = 1,2,\ldots,T\})$.

The problem (StochDPS) is a natural stochastic extension of the deterministic problem (DPS). The operational sequence of events in this stochastic model is:

(1) The state $x(t)$ is observed

(2) The decision maker selects control $v(t)$ (more precisely the function $g(t,.)$)

(3) Nature produces $\varepsilon(t+1)$

(4) State transition occurs according to $G(t,.,.,.)$

(5) Operational cycle repeats for next time period.

The analog of the larger class of sub-problems, i.e. the analog of Problem (DPSsub) (equation (40)) is:

(StochDPSsub)

$$\max_{\substack{v(t);t=t_0,1,\ldots,T-1 \\ v(t)\in\Omega(t)}} J(x_{t_0},t_0,\boldsymbol{g}) = E\{\sum_{t=t_0}^{T-1} U(t,x(t),v(t)) + S(T,x(T))\}$$

$$\text{subject to} \quad i) \quad x(t+1) = G(t,x(t),v(t),\varepsilon(t+1)), \quad t = t_0,1,\ldots,T-1, \quad (57)$$

$$ii) \quad x(t_0) = x_{t_0}, \quad \text{given.}$$

Where E denotes expectation with respect to the joint multidimensional probability mass functions (pmf) of $\{x(t); t = t_0,\ldots,T\}$ and $\{v(t); t = t_0,\ldots,T-1\}$. Or equivalently E denotes expectation with respect to the exogenous randomness at time at time t, $I_t = \left(\{x_{t_0}\},\{\varepsilon(s); s = t_0,\ldots,T\}\right)$. Then, the Bellman Equation for the stochastic problem (StochDPS) is [60, 83]:

$$\text{For} \quad k = 1,2,\ldots,T$$

$$V(x(T),0) = S(T,x(T))$$

$$V(x(T-k),k) = \max_{v(T-k)\in\Omega(T-k)} E\{U(T-k,x(T-k),v(T-k)) + V(x(T-k+1),k-1)\} \quad (58)$$

$$\text{subject to:} \quad i) \quad x(T-k+1) = G(T-k,x(T-k),v(T-k),\varepsilon(T-k+1))$$

$$ii) \quad x(T-k) = x_{T-k}, \quad \text{given.}$$

Here E denotes expectation with respect to $\varepsilon(T-k+1)$ only, as this is the only exogenous randomness for the right hand-side part of the key iteration. This fundamental iteration can be also written as (to clarify the single step optimization involved):

$$\begin{aligned}&\text{For} \quad k=1,2,...,T\\&V(x(T),0)=S(T,x(T))\\&V(x(T-k),k)=\max_{v(T-k)\in\Omega(T-k)} E\{U(T-k,x_{T-k},v(T-k))+V(G(T-k,x_{T-k},v(T-k),\varepsilon(T-k+1)),k-1)\}\end{aligned} \tag{59}$$

Because in the problems of interest in this thesis all variables can take only a finite number of values, the distinction between values of the controls from the control constraint set (i.e. $v(t)\in\Omega(t)$) and functions of the state so that their values are in the control constraint set (i.e. $g(t,.)$ such that $g(t,x(t))\in\Omega(t)$ is not needed since the two notions of control are trivially identical. As in the deterministic case, solution of this one step maximization problem, given the form of the value function from the previous step, will yield a control rule:

$$v(T-k)=g(T-k,x(T-k)),\ \ k=1,2,...,T, \tag{60}$$

that is a control law in ***state feedback form***. Exactly as in the deterministic case, after going through the successive rounds of these single period maximization problems, eventually we reach the problem at time zero, which is the original problem:

$$\begin{aligned}&V(x(0),T)=\max_{v(0)\in\Omega(0)} E\{U(0,x(0),v(0))+V(x(1),T-1)\}\\&\text{subject to:}\quad i)\quad x(1)=G(0,x(0),v(0))\\&\qquad\qquad ii)\quad x(0)=x_0,\quad \text{given,}\end{aligned} \tag{61}$$

and which will also yield the control rule:

$$v(0) = g(0, x_0). \quad (62)$$

That is $V(x_0, T) = J(x_0, \boldsymbol{g}^*)$, the optimal value of the metric, in the stochastic optimization case as well; $\boldsymbol{g}^*$ denotes the optimal policy. Knowing $x(0)$ and $v(0)$, and receiving $\varepsilon(1)$ from nature, we can use the state transition equations to compute $x(1)$. Then knowing $x(1)$ and $v(1)$, and receiving $\varepsilon(2)$ from nature, we can compute $x(2)$ and so on, until all the values of the state $x(t)$ (the state time history), and controls $v(t)$ (the controls time history) are computed. As noted already the time history of the controls is the same as the ***control policy***; recall that we denote control policies generically by $\boldsymbol{g}$. Thus, we have obtained the complete solution of the stochastic optimization problem of interest, in an efficient manner (as compared with the exhaustive evaluation of all options – like in our ECMS First Method) due to the recursive computation involved.

Section 7.3: Using DP Algorithms in the FOMCO Method

For our Second Method of performance evaluation and tradeoff analysis, FOMCO, we are thus exactly in the DP formulation we just described above. In other words if we define at each time period as "control" the pair of test and intervention applied, we have precisely the Problem (StochDPS) above (equation (56)). That is $v(t) = (\mu(t), u(t))$, for t = 0,1, 2, $N_{T,\Delta}$-1 . We now turn to the details of applying DP to compute the solution of our FOMCO, Second Method of performance evaluation and tradeoff analysis. It is easier to describe the

problem and the DP algorithm using the representation of the system dynamics via the state transition probabilities. To this end let us recall from the beginning of this section that $P(t,\mu,u)$ denotes the transition probability matrix of our CHMC, that is $P_{ij}(t,\mu,u)=\Pr\{x(t+1)=j \mid x(t)=i, v(t)=(\mu,u)\}$. In our FOMCO Method actually the underlying Markov Chain is not hidden (i.e. it is a controlled Markov chain (CMC)), as we assume we know the state at each time t (see Chapter 4 as well). Any

pair of column vectors $[X_l^{\mu}(t),\ l\in\{1,2,...,n_{\mu}\},\ t=0,1,...,N_{T,\Delta}-1]^{\mathrm{T}}$,

and $[X_k^{u}(t),\ k\in\{1,2,...,n_u\},\ t=0,1,...,N_{T,\Delta}-1]^{\mathrm{T}}$, or equivalently

$[\mu(t),u(t),t=0,1,2,...,N_{T,\Delta}-1]^{\mathrm{T}}$ is a Markov policy for any of the two optimization problems formulated; i.e. the two ways we will perform tradeoff analysis using the FOMCO method (see equations (42) and (43)). As already noted the name is completely justified as the DP algorithm will produce the values for the optimal selections of tests and interventions for each time period as functions of the state (feedback form). Recall that we denote generically by $\boldsymbol{g}$ such policies. Another representation of $\boldsymbol{g}$ is through the $2N_{\mathrm{T},\Delta}$ vector $(\boldsymbol{\mu},\boldsymbol{u})$. Let

also the n_x dimensional row vector

$$\pi_{\boldsymbol{g}}(t)=\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(\mathrm{t})=[\Pr\{x(t)=x_1 \mid \boldsymbol{\mu},\boldsymbol{u}\},\Pr\{(x(t)=x_2 \mid \boldsymbol{\mu},\boldsymbol{u}\},...,\Pr\{x(t)=x_{n_x} \mid \boldsymbol{\mu},\boldsymbol{u}\}] \tag{63}$$

denote the row vector of probabilities for the states at each time epoch (that is the state probability mass function), when policy $\boldsymbol{g}$ (or $(\boldsymbol{\mu},\boldsymbol{u})$) is used. For a

policy $\boldsymbol{g}$ (or $(\boldsymbol{\mu},\boldsymbol{u})$) we let $P_{\boldsymbol{g}}(t)$ (or $P_{(\boldsymbol{\mu},\boldsymbol{u})}(t)$) denote the corresponding state transition probability matrix at time *t*; it is a $n_x \times n_x$ matrix. From standard Markov Chain theory [45-49] we know that this probability vector under policy $\boldsymbol{g}$ (or $(\boldsymbol{\mu},\boldsymbol{u})$) evolves (i.e. dynamics of the state probability mass function) as

$$\begin{aligned} &\pi_{\boldsymbol{g}}(t+1) = \pi_{\boldsymbol{g}}(t)P_{\boldsymbol{g}}(t), \quad \pi_{\boldsymbol{g}}(0) = \pi_0 \ , \quad \text{or} \\ &\pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t+1) = \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)P_{(\boldsymbol{\mu},\boldsymbol{u})}(t), \quad \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(0) = \pi_0 \end{aligned} \tag{64}$$

where π_0 denotes the initial state probability mass function, which we take to be uniform in this thesis. The *m*-step transition probability matrix is given by

$$P_{\boldsymbol{g}}(t)P_{\boldsymbol{g}}(t+1)\ldots P_{\boldsymbol{g}}(t+m-1) \ , \tag{65}$$

and its *ij*th element is the probability that the state will be *j* at time *t + m* given that it is *i* at time *t*. Therefore

$$\begin{aligned} &\pi_g(t+m) = \pi_g(t)P_{\mathrm{g}}(t)P_{\mathrm{g}}(t+1)\ldots P_{\mathrm{g}}(t+m-1) \quad \text{and} \\ &\pi_g(t) = \pi_0 P_{\mathrm{g}}(0)P_{\mathrm{g}}(1)\ldots P_{\mathrm{g}}(t-1). \end{aligned} \tag{66}$$

Then the objective (or metric) in the problem (StochDPS) (equation (56)) can be expressed using the state transition probabilities as:

$$\begin{aligned} J(x_0,\boldsymbol{\mu},\boldsymbol{u}) &= E\{\sum_{t=0}^{N_{T,\Delta}-1} U(t,x(t),\mu(t),u(t)) + S(T,x(T))\} = \\ &= \sum_{m=1}^{m=n_x}\{\sum_{t=0}^{t=N_{T,\Delta}-1} U(t,m,\mu(t),u(t))\Pr\{x(t)=m\,|\,\boldsymbol{\mu},\boldsymbol{u}\} + S(T,m)\Pr\{x(T)=m\,|\,\boldsymbol{\mu},\boldsymbol{u}\}\}. \end{aligned}$$

(67)

Similarly for the sub-problems (equation (49)) the metric expression can be written as:

$$J(i,s,\boldsymbol{\mu},\boldsymbol{u}) = E\{\sum_{t=s}^{N_{T,\Delta}-1} U(t,x(t),\mu(t),u(t)) + S(T,x(T))\} =$$
$$= \sum_{m=1}^{m=n_x}\{\sum_{t=s}^{t=N_{T,\Delta}-1} U(t,m,\mu(t),u(t))\Pr\{x(t)=m\,|\,\boldsymbol{\mu},\boldsymbol{u}\} + S(T,m)\Pr\{x(T)=m\,|\,\boldsymbol{\mu},\boldsymbol{u}\}\},$$

(68)

for $i = 1,2,...,n_x$, where we have taken $x(0) = x_i$ and $t_0 = s$, in $J(x_{t_0}, t_0, \boldsymbol{g})$ (eq. (49)). That is $J(i,s,\boldsymbol{\mu},\boldsymbol{u})$ is the expected value of the objective when using the Markov policy $\boldsymbol{g} = (\boldsymbol{\mu},\boldsymbol{u})$, and starting at time *s* from state *i,* and running till time $N_{T,\Delta}$. It is often called ***"the cost to go"***, a terminology coming from optimization problems where the objective is also called "the cost" [60]. Given a policy $(\boldsymbol{\mu},\boldsymbol{u})$ the value of the metric $J(i,s,\boldsymbol{\mu},\boldsymbol{u})$ can be calculated by the backwards recursion:

$$J(i,s,\boldsymbol{\mu},\boldsymbol{u}) = U(s,i,g(s,i)) + \sum_{j=1}^{n_x}(P_{(\mu,\boldsymbol{u})}(s))_{ij} J(j,s+1,\boldsymbol{\mu},\boldsymbol{u}), \quad s = N_{T,\Delta}-1,...,2,1,0, \quad \text{for all } i$$
$$J(i,N_{T,\Delta},\boldsymbol{\mu},\boldsymbol{u})) = S(N_{T,\Delta},i), \quad \text{for all } i. \tag{69}$$

Often we want to evaluate the performance of a stochastic system regardless of the initial state. That is we want to compute $E_{x_0}\{J(x_0,\boldsymbol{\mu},\boldsymbol{u})\}$, where E_{x_0} indicates the expectation with respect to the statistics of the initial state condition x_0. If $J_{(\mu,\boldsymbol{u})}$ denotes the corresponding expected value of the objective or metric, it can be easily computed as

$$J_{(\mu,\boldsymbol{u})} = \sum_{m=1}^{n_x} J(m,0,\boldsymbol{\mu},\boldsymbol{u})\Pr\{x(0)=m\}. \tag{70}$$

The above fundamental recursion can be represented in a vector-matrix notation (which is also useful for the MATLAB implementation of the algorithm) as follows:

$$
\begin{aligned}
&J_{\boldsymbol{g}}(s) = f_{\boldsymbol{g}}(s) + P_{\boldsymbol{g}}(s) J_{\boldsymbol{g}}(s+1), \quad s = N_{T,\Delta} - 1, \ldots, 2, 1, 0 \\
&J_{\boldsymbol{g}}(N_{T,\Delta}) = f_{\boldsymbol{g}}(N_{T,\Delta}) \\
&J_{\boldsymbol{g}} = \pi_0 J_{\boldsymbol{g}}(0)
\end{aligned} \tag{71}
$$

where $J_{\boldsymbol{g}} = J_{(\mu,\boldsymbol{u})}(s) = [J(x_1, s, \boldsymbol{\mu}, \boldsymbol{u}), J(x_2, s, \boldsymbol{\mu}, \boldsymbol{u}), \ldots, J(x_{n_x}, s, \boldsymbol{\mu}, \boldsymbol{u})]^T$, an n_x - dimensional vector,

$f_{\boldsymbol{g}}(s) = f_{(\mu,\boldsymbol{u})}(s) = [U(s, x_1, g(s, x_1)), U(s, x_2, g(s, x_2)), \ldots, U(s, x_{n_x}, g(s, x_{n_x}))]^T$, for $s = 0, \ldots, N_{T,\Delta} - 1$ an n_x - dimensional vector,

$f_g(N_{T,\Delta})(i) = f_{(\mu,\boldsymbol{u})}(N_{T,\Delta})(i) = S(N_{T,\Delta}, i)$, for all i , and $\boldsymbol{g} = (\boldsymbol{\mu}, \boldsymbol{u})$ is the Markov policy applied.

This vector-matrix notation is useful also (for the same reasons) in representing the fundamental Bellman recursion of DP. Indeed the optimal value of the metric $J_{\boldsymbol{g}} = J_{(\mu,\boldsymbol{u})}$ is determined by the backwards recursion:

$$
\begin{aligned}
&J_{(\mu,\boldsymbol{u})}(N_{T,\Delta}) = f_{(\mu,\boldsymbol{u})}(N_{T,\Delta}) \\
&J_{(\mu,\boldsymbol{u})}(s) = \max_{(\mu,u)\ \text{allowed}} \{ f_{(\mu,\boldsymbol{u})}(s) + P_{(\mu,\boldsymbol{u})}(s) J_{(\mu,\boldsymbol{u})}(s+1) \} \\
&J_{(\mu,\boldsymbol{u})} = \pi_0 J_{(\mu,\boldsymbol{u})}(0)
\end{aligned} \tag{72}
$$

where the maximum is taken **separately for each component of this vector equation** and over the allowed (by the constraints, μ and u).

We can now apply these general results to our problem. We first develop the DP-based algorithm for the first technique of computing Pareto points using the FOMCO method, the FOMCO-SN method, that combines the two normalized metrics in a convex combination (eq. (42)). The state transition probability

matrix for our problem is time invariant; it does not depend explicitly on *t.* It is a 3×3 stochastic matrix described in our model in Section 4.1. This is our $P_{\boldsymbol{g}} = P_{(\boldsymbol{\mu},\boldsymbol{u})}$. Further for our model $\pi_0 = [1/3, 1/3, 1/3]$; i.e. uniform. The analog of $J_{(\boldsymbol{\mu},\boldsymbol{u})}$ in equation (72) above, is

$$
\begin{aligned}
J_{cc}^{\lambda}(p_i,\boldsymbol{\mu},\boldsymbol{u}) &= \lambda \bar{J}_{hc}^{r,n}(p_i,\boldsymbol{\mu},\boldsymbol{u}) + (1-\lambda)(-C^{total,n}(p_i,\boldsymbol{\mu},\boldsymbol{u})) \\
&= \lambda \sum_{t=0}^{N_{T,\Delta}} \sum_{m=1}^{3} (V_m^{p_i} / J_n^r) \Pr\{x(t)=m \mid \boldsymbol{\mu},\boldsymbol{u}\} + (1-\lambda)(-(\sum_{t=0}^{N_{T,\Delta}-1} C_{\mu}(\mu(t)) + \sum_{t=0}^{N_{T,\Delta}-1} C_u(u(t)))/C_n) \\
&= \sum_{t=0}^{N_{T,\Delta}-1} (\lambda \sum_{m=1}^{3} (V_m^{p_i} / J_n^r)\, \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m + (\lambda-1)(C_{\mu}(\mu(t)) + C_u(u(t)))/C_n) + \lambda \sum_{m=1}^{3} (V_m^{p_i} / J_n^r)\, \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(N_{T,\Delta})_m \qquad (73) \\
&= \sum_{t=0}^{N_{T,\Delta}-1} (\lambda E\{V_{x(t)}^{p_i} / J_n^r\} + (\lambda-1)(C_{\mu}(\mu(t)) + C_u(u(t)))/C_n + \lambda E\{V_{x_{(N_{T,\Delta})}}^{p_i} / J_n^r\} \\
&= E\{\sum_{t=0}^{N_{T,\Delta}-1} (\lambda (V_{x(t)}^{p_i} / J_n^r) + (\lambda-1)(C_{\mu}(\mu(t)) + C_u(u(t)))/C_n + \lambda (V_{x_{(N_{T,\Delta})}}^{p_i} / J_n^r)\}
\end{aligned}
$$

where p_i denotes patient *i* (to avoid notation confusion with state *i*). Note also, again, that the computation is different for the various patient types -- the *r* superindex. We note that the expectation in the above equation includes expectation with respect to the statistics of the initial state x_0. We note that our problem is simpler than the general stochastic optimization problem described earlier in relation to Dynamic Programming (see problem (StochDPS) above) : the state transition probability matrix does not depend explicitly on time; the objective function (metric) does not depend explicitly on time, neither does the final time metric; the constraints on tests and interventions are simple (there are three possible tests and ten possible interventions). By comparing equations (eq. (56)) and (eq. (73)) we can make the following identifications between our

specific problem and the general formulation of the DP algorithms for the stochastic optimization problem (StochDPS):

$$
\begin{aligned}
&J_n^r = (11\,(V_{\max}^r - V_{\min}^r)), \text{ for } r = 1,2,3\\
&C_n = (10\,((c_{\max}^{\mu} - c_{\min}^{\mu}) + (c_{\max}^{u} - c_{\min}^{u})))\\
&S(T, x(T)) = S^{\lambda}(x(T)) = \lambda V_{x(T)}^{p_i} / J_n^r\\
&U(t, x(t), v(t)) = U^{\lambda}(x(t), v(t)) = \lambda V_{x(t)}^{p_i} / J_n^r + (\lambda - 1)(C_{\mu}(\mu(t)) + C_u(u(t)) / C_n\\
&v(t) = (\mu(t), u(\mathrm{t}))\\
&\mu(t) \in \{\mu_1, \mu_2, \mu_3\}\\
&u(t) \in \{u_1, u_2, \ldots, u_{10}\}\\
&C_{\mu}(\mu_l) = c_l^{\mu}, \quad l = 1,2,3\\
&C_u(u_k) = c_k^{u}, \quad k = 1,2,3,\ldots,10\\
&P_{(\mu,u)} = \text{ as given in Section 4.1}\\
&\pi_0 = [1/3, 1/3, 1/3]\\
&T = 10
\end{aligned}
\tag{74}
$$

With these identifications the fundamental backwards DP recursion leads to the following computational algorithm, which we call ***Algorithm 2a – tradeoff-via-FOMCO-SN.***

Algorithm 2a -- tradeoff-via-FMCO-SN

With the definitions of equations (74)

For $\lambda = 0.1, 0.2, ..., 0.9$

Define

for $s = 0, 1, ..., 9$

$J_{\boldsymbol{g}} = J_{\boldsymbol{g}}^{\lambda} = J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(s) = [J(1, s, \boldsymbol{\mu}, \boldsymbol{u}), J(2, s, \boldsymbol{\mu}, \boldsymbol{u}), J(3, s, \boldsymbol{\mu}, \boldsymbol{u})]^T$, a 3D vector

$f_{\boldsymbol{g}}(s) = f_{\boldsymbol{g}}^{\lambda}(s) = f_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(s) = [U(s, 1, g(s,1)), U(s, 2, g(s,2)), U(s, 3, g(s,3))]^T$, a 3D vector

$f_{\boldsymbol{g}}(10)(i) = f_{\boldsymbol{g}}^{\lambda}(10)(i) = f_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(10)(i) = S(10, i) = S^{\lambda}(i)$, for $i = 1, 2, 3$

$\boldsymbol{g} = (\boldsymbol{\mu}, \boldsymbol{u})$

Next run the (3D vector) backwards iterations below for $s = 9, 8, ..., 1, 0,$

$J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(10) = f_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(10)$

$J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(s) = \max_{(\mu,u)\text{ allowed}} \{ f_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(s) + P_{(\boldsymbol{\mu},\boldsymbol{u})} J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(s+1) \}$, component-wise max

Next compute

$J_{(\boldsymbol{\mu},\boldsymbol{u})} = J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda} = \pi_0 J_{(\boldsymbol{\mu},\boldsymbol{u})}^{\lambda}(0)$

Use the computed optimal sequences of tests $\mu^{\lambda,*}(s)$, interventions $u^{\lambda,*}(s)$, $s = 0, 1, 2, ..., 9$ and states $x^{\lambda,*}(s)$, $s = 0, 1, 2, ..., 10$ to compute the values of the two metrics

$$\bar{J}_{hc}(p_i, \boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*}) = \sum_{t=0}^{9} \sum_{m=1}^{3} V_m^{p_i} \pi_{(\boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*})}(t)_m + \sum_{m=1}^{3} V_m^{p_i} \pi_{(\boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*})}(10)_m \tag{75}$$

$$C^{total}(p_i, \boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*}) = \sum_{t=0}^{9} (C_{\mu}(\mu^{\lambda,*}(t)) + C_u(u^{\lambda,*}(t)))$$

Plot the point $(\bar{J}_{hc}(p_i, \boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*}), C^{total}(p_i, \boldsymbol{\mu}^{\lambda,*}, \boldsymbol{u}^{\lambda,*}))$ in the 2D tradeoff plane

Repeat for the next value of λ

Here we have introduced the superscript λ to emphasize the dependence of these functions on the parameter λ. As we step through the values of λ the algorithm computes Pareto points. We can compute more Pareto points if we select a finer quantization for λ; like for example $\lambda = 0.01, 0.02, ..., 0.99$.

We next develop a DP-based algorithm for the second method of computing the Pareto points, as described in equations (43) for FOMCO-E. The representations

of the two metrics $\bar{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ and $C^{total}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ are the same as in the first method above. In this second method we will step through a sequence of constraints for $C^{total}(p_i, \boldsymbol{\mu}, \boldsymbol{u}) \le \varepsilon$, for the chosen values of ε, which will result in constraints on the sequences of tests $\{\mu(s), s = 0,1,2,...,9\}$ and interventions $\{u(0), u(1), u(2), ..., u(9)\}$ we can use. Let us denote the sets of allowed tests and interventions $\Omega_{\mu}^{\varepsilon}(s)$ and $\Omega_{u}^{\varepsilon}(s)$ respectively, where we have allowed these sets to depend on time to cover all possibilities. The algorithm needs to describe how these sets are computed first using the given information:

$$\begin{aligned} &\mu(t) \in \{\mu_1, \mu_2, \mu_3\} \\ &u(t) \in \{u_1, u_2, ..., u_{10}\} \\ &C_{\mu}(\mu_l) = c_l^{\mu}, \quad l = 1,2,3 \\ &C_u(u_k) = c_k^u, \quad k = 1,2,3,...,10 \end{aligned} \tag{76}$$

Then we will compute the sequences of tests and interventions maximizing the metric $\bar{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u})$ subject to these constraints. This maximization problem is also a problem of the type (StochDPS) since it is described as

$$\max_{\substack{(\mu(t),u(t)),\, t=0,1,2,...,9 \\ \mu(t) \in \Omega_{\mu}^{\varepsilon}(t),\ u(t) \in \Omega_{u}^{\varepsilon}(t)}} \bar{J}_{hc}(p_i, \boldsymbol{\mu}, \boldsymbol{u}) = E\{\sum_{t=0}^{9} V_{x(t)}^{p_i} + V_{x_{(10)}}^{p_i}\} = \sum_{t=0}^{9} (\sum_{m=1}^{3} V_m^{p_i} \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(t)_m) + \sum_{m=1}^{3} V_m^{p_i} \pi_{(\boldsymbol{\mu},\boldsymbol{u})}(10)_m \,. \tag{77}$$

By comparing equations (56) and (77) we can make the following identifications between our specific problem and the general formulation of the DP algorithms for the stochastic optimization problem (StochDPS):

$$
\begin{aligned}
&S(T, x(T)) = S(x(T) = V_{x(10)}^{p_i} \\
&U(t, x(t), v(t)) = U(x(t), v(t)) = V_{x(t)}^{p_i} \\
&v(t) = (\mu(t), u(\mathrm{t})) \\
&\mu(t) \in \Omega_{\mu}^{\varepsilon}(t) \\
&u(t) \in \Omega_{u}^{\varepsilon}(t) \\
&P_{(\mu,u)} = \text{ as given in Section 4.1} \\
&\pi_0 = [1/3, 1/3, 1/3] \\
&T = 10
\end{aligned}
\tag{78}
$$

These identifications and the fundamental backwards DP recursion lead to the following computational algorithm, which we call ***Algorithm 2b – tradeoff-via-FOMCO-E.***

As we step through the values of ε the algorithm computes Pareto points. We can compute more Pareto points if we select a finer quantization for ε.

As Algorithm 2a is much more efficient and less computationally costly than Algorithm 2b, we did not implement Algorithm 2b, neither we performed experiments with it.

Algorithm 2b – tradeoff-via-FOMCO-E

With the definitions of equations (74)

For $\varepsilon = 2,600,\ 2,700,\ 2,800,...,\ 6,000$

Define

for $s = 0,1,...,9$

$J_{\boldsymbol{g}} = J_{(\mu,\boldsymbol{u})}(s) = [J(1,s,\boldsymbol{\mu},\boldsymbol{u}), J(2,s,\boldsymbol{\mu},\boldsymbol{u}), J(3,s,\boldsymbol{\mu},\boldsymbol{u})]^T$, a 3D vector

$f_{\boldsymbol{g}}(s) = f_{(\mu,\boldsymbol{u})}(s) = [U(s,1,g(s,1)), U(s,2,g(s,2)), U(s,3,g(s,3))]^T$, a 3D vector

$f_{\boldsymbol{g}}(10)(i) = f_{(\mu,\boldsymbol{u})}(10)(i) = S(10,i)$, for $i = 1,2,3$

$\boldsymbol{g} = (\boldsymbol{\mu},\boldsymbol{u})$

Next run the (3D vector) backwards iterations below for $s = 9,8,...,1,0,$

$J_{(\mu,\boldsymbol{u})}(10) = f_{(\mu,\boldsymbol{u})}(10)$

Compute $\Omega_{\mu}^{\varepsilon}(s), \Omega_{u}^{\varepsilon}(s)$, for $s = 8,...,1,0$, as $\{\mu_1, \mu_2, \mu_3, u_1, u_2, ..., u_{10}$, such that

$$\sum_{t=0}^{s}(C_{\mu}(\mu(t)) + C_u(u(t))) \le \varepsilon - \sum_{t=s+1}^{9}(C_{\mu}(\mu^{\varepsilon,*}(t)) + C_u(u^{\varepsilon,*}(t))),$$

with $\mu(t) \in \{\mu_1, \mu_2, \mu_3\},\ u(t) \in \{u_1, u_2, ..., u_{10}\},$

$\mu^{\varepsilon,*}(t), u^{\varepsilon,*}(t), t = s+1,...,9,$ the maximizers of previous step

$$J_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})}(s) = \max_{(\mu(s)\in\Omega_{\mu}^{\varepsilon}(s),\, u(s)\in\Omega_{u}^{\varepsilon}(s))} \{f_{(\mu,u)}(s) + P_{(\mu,u)}J_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})}(s+1)\},$$

with $\mu^{\varepsilon,*}(s), u^{\varepsilon,*}(s)$ denoting the maximizing variables

Next compute

$J_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})} = \pi_0 J_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})}(0)$

Use the computed sequences of tests $\mu^{\varepsilon,*}(s)$, interventions $u^{\varepsilon,*}(s)$, $s = 0,1,2,...,9$ and states $x^{\varepsilon,*}(s)$, $s = 0,1,2,...,10$ to compute the values of the two metrics

$$\overline{J}_{hc}(p_i, \boldsymbol{\mu}^{\varepsilon,*}, \boldsymbol{u}^{\varepsilon,*}) = \sum_{t=0}^{9}\sum_{m=1}^{3} V_m^{p_i} \pi_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})}(t)_m + \sum_{m=1}^{3} V_m^{p_i} \pi_{(\mu^{\varepsilon,*},\boldsymbol{u}^{\varepsilon,*})}(10)_m$$

$$C^{total}(p_i, \boldsymbol{\mu}^{\varepsilon,*}, \boldsymbol{u}^{\varepsilon,*}) = \sum_{t=0}^{9}(C_{\mu}(\mu^{\varepsilon,*}(t)) + C_u(u^{\varepsilon,*}(t)))$$

Plot the point $(\overline{J}_{hc}(p_i, \boldsymbol{\mu}^{\varepsilon,*}, \boldsymbol{u}^{\varepsilon,*}), C^{total}(p_i, \boldsymbol{\mu}^{\varepsilon,*}, \boldsymbol{u}^{\varepsilon,*}))$ in the 2D tradeoff plane

Repeat for the next value of ε

Section 7.4: Output and Results from Algorithm 2a--trade-off-via-FOMCO-SN

The implemented Second Method is Algorithm 2a—tradeoff-via-FOMCO-SN. Our MBSE system outputs directly the Pareto-points and other related information. This algorithm is very fast. For the same problems that the First Method (EMCS) took 783 sec (two metrics, 10,000 patients, 32 runs) and 1,385 sec (3 metrics, 100,000 patients, 9 runs), our Second Method (FOMCO-SN) took only 2.36 sec on the same laptop.

We provide below three examples of the produced graphs, one each for each type of patient. The values of λ we used were λ = 0.005, 0.01, 0.015, …, 0.995; that is 200 values. The Pareto points computed and Pareto frontiers computed are very similar to those computed with our First Method (EMCS) but at a fraction of the time!

We note that the horizontal set of points at the top of each graph are erroneous Pareto points due to the underlying sampling of the values of λ, and the fact that our underlying decision problem is actually discrete, more like a stochastic integer programming problem. These phenomena appear in scalarization methods as scalarization methods are excellent for continuous problems but more challenging for discrete problems like ours. This problem is easily fixed by running a (discrete) Pareto points algorithm [17-21] at the end of the scalarization algorithm. Indeed we tried this and the horizontal lines are eliminated.

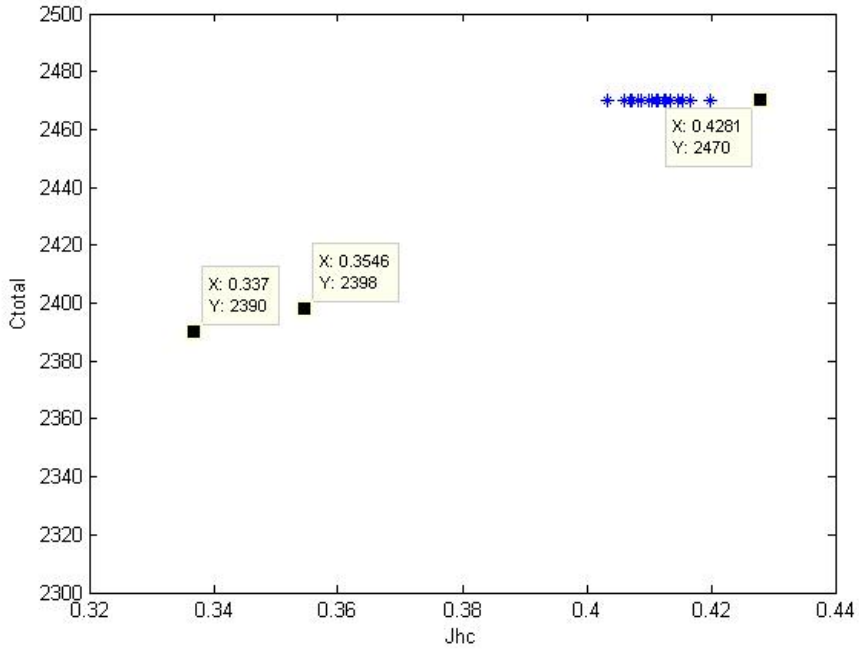


Fig. 22: Typical 2-D graph produced by our MBSE system, FOMCO-SN Method, that gives the Pareto frontier for a typical risk averse patient (black squares are the Pareto points).

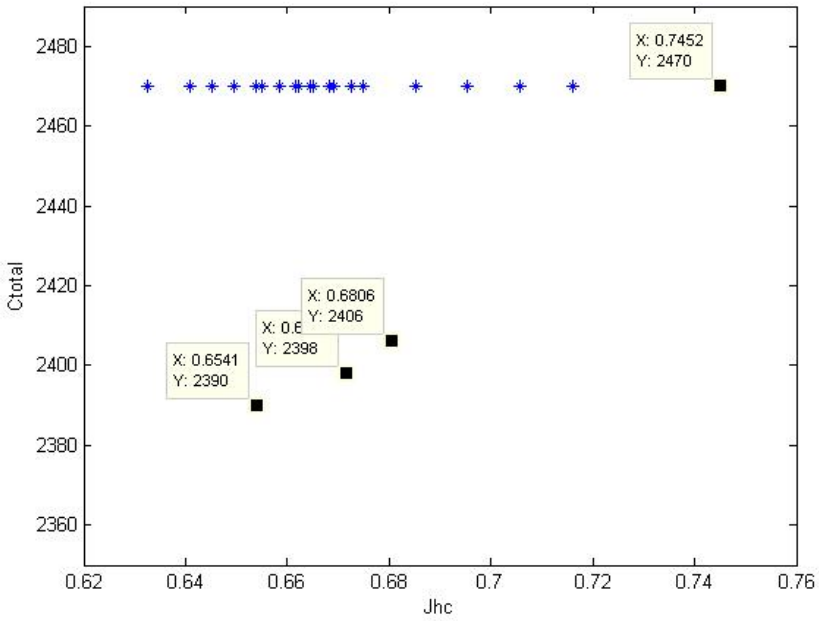


Fig. 23: Typical 2-D graph produced by our MBSE system, FOMCO-SN Method that gives the Pareto frontier for a typical risk indifferent patient (black squares are the Pareto points).

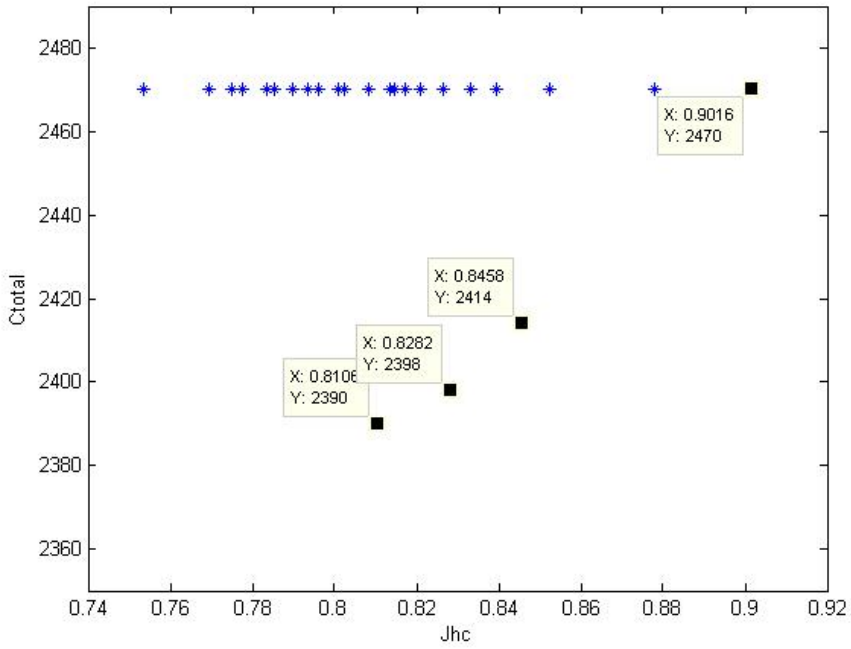


Fig. 24: Typical 2-D produced by our MBSE system, FOMCO-SN Method that gives the Pareto frontier for a typical risk taker patient (black squares are the Pareto points).

Chapter 8: **Model Based Systems Engineering for Health Care Management Systems for Diabetes Mellitus**

In this chapter we provide some illustrative examples about applying MBSE, as described in Chapter 2, for the fundamental components, design and operation of a Health Care Management System (HCMS) of the type of interest in this thesis. We will use a HCMS for Diabetes Mellitus as the special case of interest and focus.

Section 8.1: Scope of MBSE for HMSC for Diabetes Mellitus

Modern and future Health Care Information Technology [1, 7, 14, 15, 16] will have high level of connectivity, increased amounts of different types of data and will support smart and informed responses to queries by heterogeneous users (with different expertise and knowledge in medicine and health care). The high level environment is called a Health Information Exchange [HIE] system that performs as a digital highway for the traffic of health information. Underneath the HIE stand several subsystems that communicate with each other. These systems are Electronic Health Records [EHR] that stands inside the medical and nurse practitioner data silos, the personal health records that has vital signs and test results for every patient and are stored in labs and patients personal databases, and finally Electronic Medical Records [EMR] as a main operational system that manages information of patients and at the same time produces

valuable reports for the decision maker. These Health IT systems and components are essential for current and future Health Care Management Systems (HCMS) as discussed in Chapter 1 of this thesis (see also Figure 1).

We assume that, hopefully, the Meaningful Use Act II and III [1] come in force and that the health care actors agree accordingly to implement them. These policies and events create a tremendous opportunity to implement and enact a Model Based System Engineering approach for Decision Making in Chronic Diseases, and in our case will be Diabetes Type Two. The proposed system will operate "underneath" the above mentioned systems, who extract and store information. The system has a main objective to receive information from the potential health policy actors and perform **tradeoff analysis** and respond to high level **"what if"** questions and queries that will be demanded by health care professionals, medical facilities, medical insurers and even individual patients [16]. Such HMCS would receive information about diagnostic tests, performed interventions, period of therapy and probabilities from transitions from each health state (clinical studies, systematic reviews, institute guidelines) and interactively (with the users) perform tradeoff analysis and plot / visualize the tradeoff results, based on optimization models which choose for their analysis. The therapy period we assume is ten years, the number of potential interventions is ten and the number of the potential test is three more and detailed information will be provided letter in this thesis. For the transition probability matrix the policy maker has run a clinical study and has an epidemiological data that shows the potential transitions between health states.

Section 8.2: Use Case Diagram for HCMS for Diabetes Mellitus

In Figure 25 we give a high level illustration of a use case diagram that illustrates how the actors and the Health Care Management System (HCMS) interact. As discussed in Chapter 2 of this thesis, Use Case Diagrams are a fundamental tool in MBSE [17-20], and are systematically used to generate use cases, link the system requirements to models of systems structure and behavior, and subsequently to design space exploration and tradeoffs, and finally to system validation-verification-testing (see Chapter 2 of this thesis).

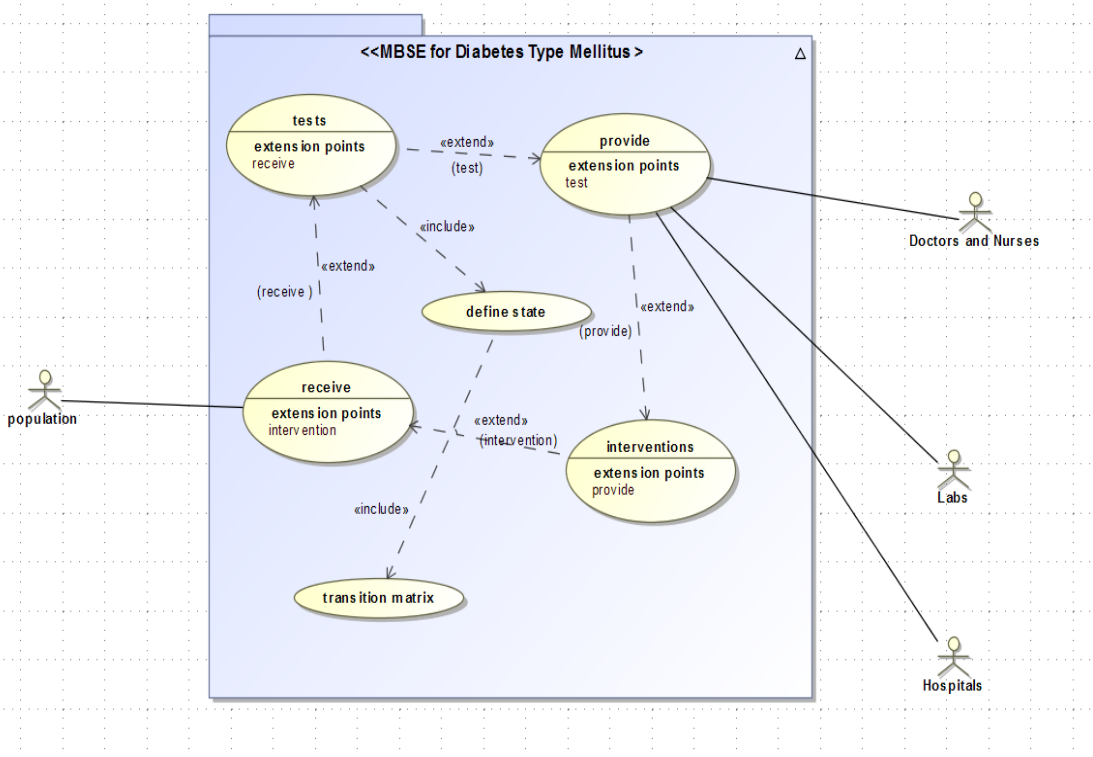


Fig. 25: Use Case Diagram for a HCMS for Diabetes Type two (Diabetes Mellitus)

Section 8.3: Use Cases Index and Descriptions for a HCMS for Diabetes Mellitus

In this section we provide examples of use cases for the HCMS for Diabetes Mellitus and their descriptions as they form a fundamental component of MBSE [17-20]. This is included here for illustrative purposes and for completeness.

Use case ID	Use Case Name	Primary Actor	Scope	Complexity
1	Receive test	Population	in	High
2	Receive intervention	population	in	high
3	Provide test	lab	in	med
4	Provide intervention	Doctors nurses	in	med
5	Provide transition metric	hospital	in	high

Identification of each key component of your use case

Use case element	Description
Use case number	1
Application	MBSE for Diabetes Type two (Mellitus)
Use case Name	Patient receive test
Use case Description	The patient receives one of the three available tests for identification of the level of sugar in his blood as diagnostic for Diabetes Mellitus
Primary Actor	Population
Precondition	The patient was recommended to receive test from a physician
Trigger	Social and clinical risk factors act as a trigger for a member of the general population to receive a diagnostic test
Basic Flow	1. The patient receives a notification based on her risk factors to receive a test from her doctor 2. The doctor informs the risk factors and the prescription to a hospital system 3. The hospital informs the labs that they will receive a patient for a specific test 4. The patient visits the lab and receives the test 5. The lab reads and prints results and send them to the doctor 6. Doctor decides to prescribe an intervention 7. Hospital has being informed for the specific patient

	intervention and informs the transition state probabilities 8. Patient is being consulted for following an intervention for period of time (therapy period)
Alternate Flows	1. The patient performs a general check-up routine in a local lab 2. The patient informs his personal doctor about his lab results 3. The doctor suggests an intervention for a time period 4. Patient visits the local hospital with doctor's suggestive therapy and receives the intervention 5. Hospital, doctor and lab periodically update patient's health information about his state and her probabilities of health state transition
Use case element	Description
Use case number	2
Application	MBSE for Diabetes Type two (Mellitus)
Use case Name	Patient receives an intervention
Use case Description	The patient receives one of the ten available interventions for reducing the level of sugar in his blood, as a result of positive diagnosis of Diabetes Mellitus
Primary Actor	Population
Precondition	The patient was recommended to receive an intervention from a physician
Trigger	The positive diagnostic test results for Diabetes Mellitus have triggered the provision of a focused intervention
Basic Flow	1. Patient receives proposed intervention 2. Patient follows the behavioral intervention 3. Patient purchases, or is being provided with, equipment that will survey his state and will inform his assigned nurse 4. Nurse receives the daily messages and writes reports that will be sent to the doctor 5. The doctor based on the yearly results, asks for a second diagnostic test and continues or changes the intervention 6. Doctor informs hospital about his actions regarding tests and interventions by following standard guidelines 7. Hospital calculates the results based on tradeoff analysis and suggests alternatives
Alternate Flows	1. Patient receives an intervention

	2. Patient follows the behavioral intervention 3. Patient visits the hospital on a regular basis 4. In the hospital they measure patient's vital signs 5. At the end of the year period the information from the hospital is send to patient's personal doctor 6. Patient's personal doctor prescribes a test 7. Based on the results (health state) patient follows the same or different intervention

Use case element	Description
Use case number	3
Application	MBSE for Diabetes Type two (Mellitus)
Use case Name	Lab provides test
Use case Description	The lab after being informed about the prescription by a doctor performs the suggested test
Primary Actor	Lab
Precondition	The lab is adequately prepared in facilities, equipment, tests, and personnel to receive population for Diabetes Mellitus tests
Trigger	Lab receives a prescription from doctor or hospital to perform a specific test
Basic Flow	1. Lab receives a prescription from doctor to perform a test to patient 2. Lab sets an appointment with patient and suggests to him fasting for a period before he will receive the test 3. Patient receives the test 4. Patient receives the results of the test
Alternate Flows	1. Lab receives prescription from hospital department 2. Patient receives pre-test consultation and arrangements for appointment 3. Patient receives the test 4. Lab informs hospital about the results of the test.

Use case element	Description
Use case number	4
Application	MBSE for Diabetes Type two (Mellitus)
Use case Name	Doctor Provides Intervention
Use case Description	Doctor suggests test and based on the interpretation of the test results suggests an intervention

Primary Actor	Doctor
Precondition	Doctor is informed about Diabetes Mellitus health states, and has received information about the transition probabilities matrix
Trigger	The results from Diabetes Mellitus test are positive and the state of the patient results in a recommendation to receive one of the 10 suggested interventions
Basic Flow	1. Doctor receives test results from patient and the results map the patient's health state in the 2nd or 3rd state 2. Doctor consults the guidelines, the health state definition and the state transition matrix and suggests an intervention 3. Doctor informs hospital for the period of the therapy 4. Doctor assigns a nurse to patient for a period until the next screening test period for the patient
Alternate Flows	1. Hospital receives the results from lab 2. Hospital assigns a doctor to interpreting the results, consult the patient 3. Doctor makes a decision and informs the patient to follow a specific intervention 4. Doctor informs hospital about his health strategy

Use case element	Description
Use case number	5
Application	MBSE for Diabetes Type two (Mellitus)
Use case Name	Hospital provides state transition probability matrix
Use case Description	An important component for the overall therapy is the definition of the state of the disease (that is provided by Medical Authority) and the state transition probabilities that are constructed collaboratively between the hospital and other authorities
Primary Actor	Hospital
Precondition	Hospital receives information from labs and doctors about success or failure from pairs of (intervention and test) and stores them in a data base
Trigger	Call from a doctor in order to suggest a therapy for transition matrix
Basic Flow	1. Doctor requests state transition probability matrix 2. Doctor is informed about how to set up a query 3. The query selects and constructs a table with the needed health state transition probabilities

	4. Doctor receives the state transition matrix and inputs it for the suggested therapy 5. Hospital records doctor's query and updates the knowledge base 6. Hospital database performs a group order to construct arrays of triples, consisting of state, test and intervention 7. Hospital keeps the counting statistics 8. Hospital assigns cost, health performance and runs tradeoff analysis
Alternate Flows	1. The hospital receives information about a test and its results 2. Hospital inquires if the test was connected with an intervention 3. Hospital based on the therapy period, test and intervention constructs state transition probability matrix 4. Hospital provides access to the state transition matrix to a doctor or nurse 5. Hospital runs periodically tradeoff analyses and informs the other actors via guidelines for excellence (in treatment)

Section 8.4: Structure (Block) Diagram of a HCMS for Diabetes II

As discussed in Chapter 2, a fundamental construct in MBSE is the Structure (or Block) Diagram [17-20]; which is one of the four pillars of SysML [17-20] and provides a model of the **System Structure** (i.e. it answers the question "what the system consists of?"). In Figure 26 we provide the Structure Diagram for a HCMS for Diabetes Mellitus; the diagram also provides a hierarchical representation of the HCMS for Diabetes Mellitus. The HCMS begins with a Health Information exchange system, continues with several individual subsystems of Electronic Health Records and Electronic Medical Records and finishes with the Reasoning Engine (designated as MBSE system in the diagram) of the HMCS for Diabetes Mellitus; the component which is the main focus of most of the work described

in this thesis. The Reasoning Engine (MBSE system) subsystem of the HCMS for Diabetes Mellitus, is comprised of three main subsystems: the EMCS, the FOMCO and the POMCO. The detailed operation of the EMCS subsystem was provided in Chapter 6 of this thesis. The detailed operation of the FOMCO subsystem was provided in Chapter 7 of this thesis. The detailed operation of POMCO has been left as future research topic; a brief description was given in Subsection 4.2.2, pages 44-45, of this thesis, and more can be found in our forthcoming paper [87]. These subsystems receive the same inputs, which are typically communicated on the basis of counting statistics for various relevant variables. The inputs of cost, health state, risk behavior and probability matrix of patient health state transitions are defined by the actors and other databases. In Figure 27 we provide the Structure Diagram for a part (the information flow of the EMCS subsystem) of the Reasoning Engine of the HCMS for Diabetes Mellitus.

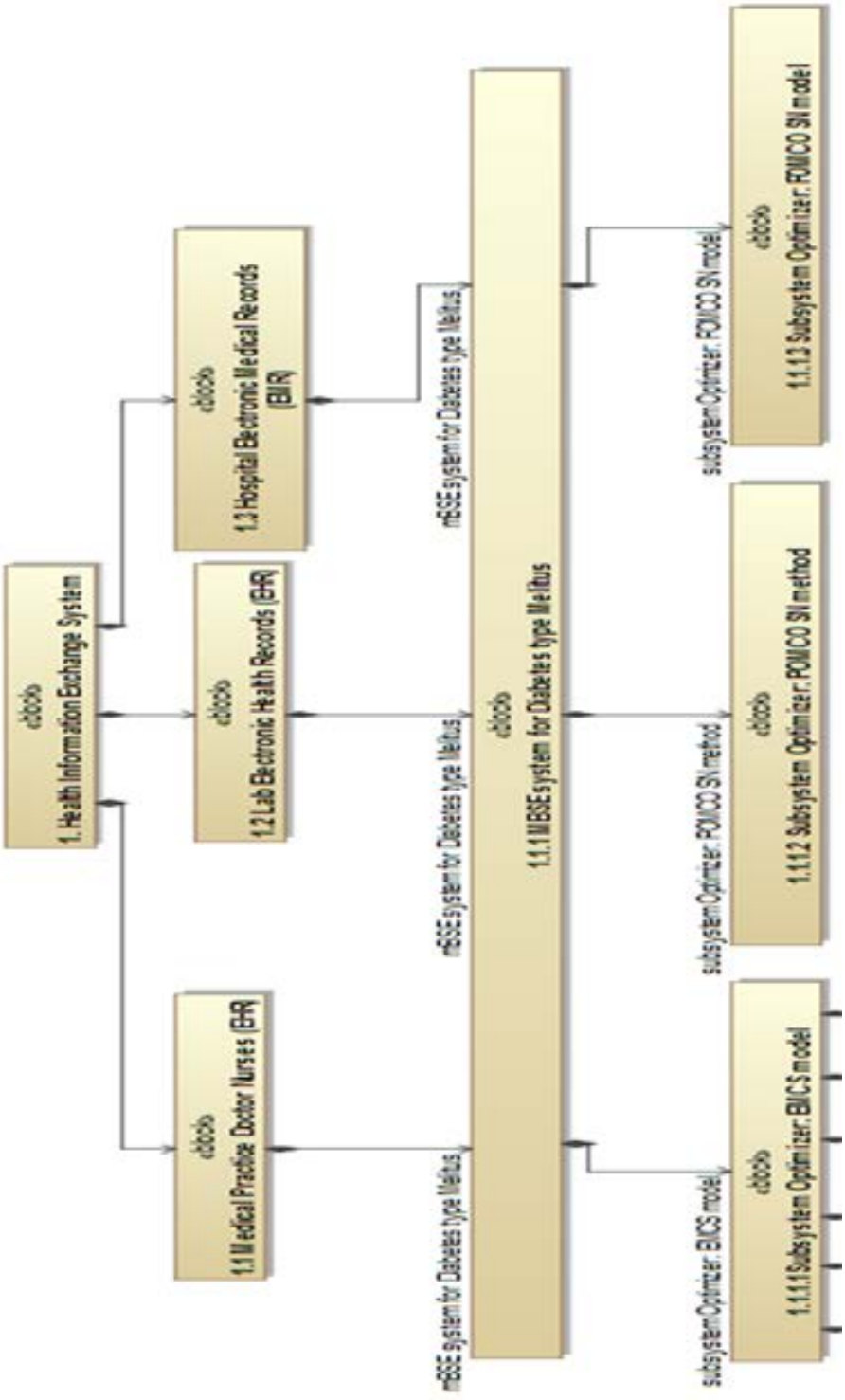


Fig. 26: Structure (Block definition) Diagram for the HCMS for Diabetes Mellitus

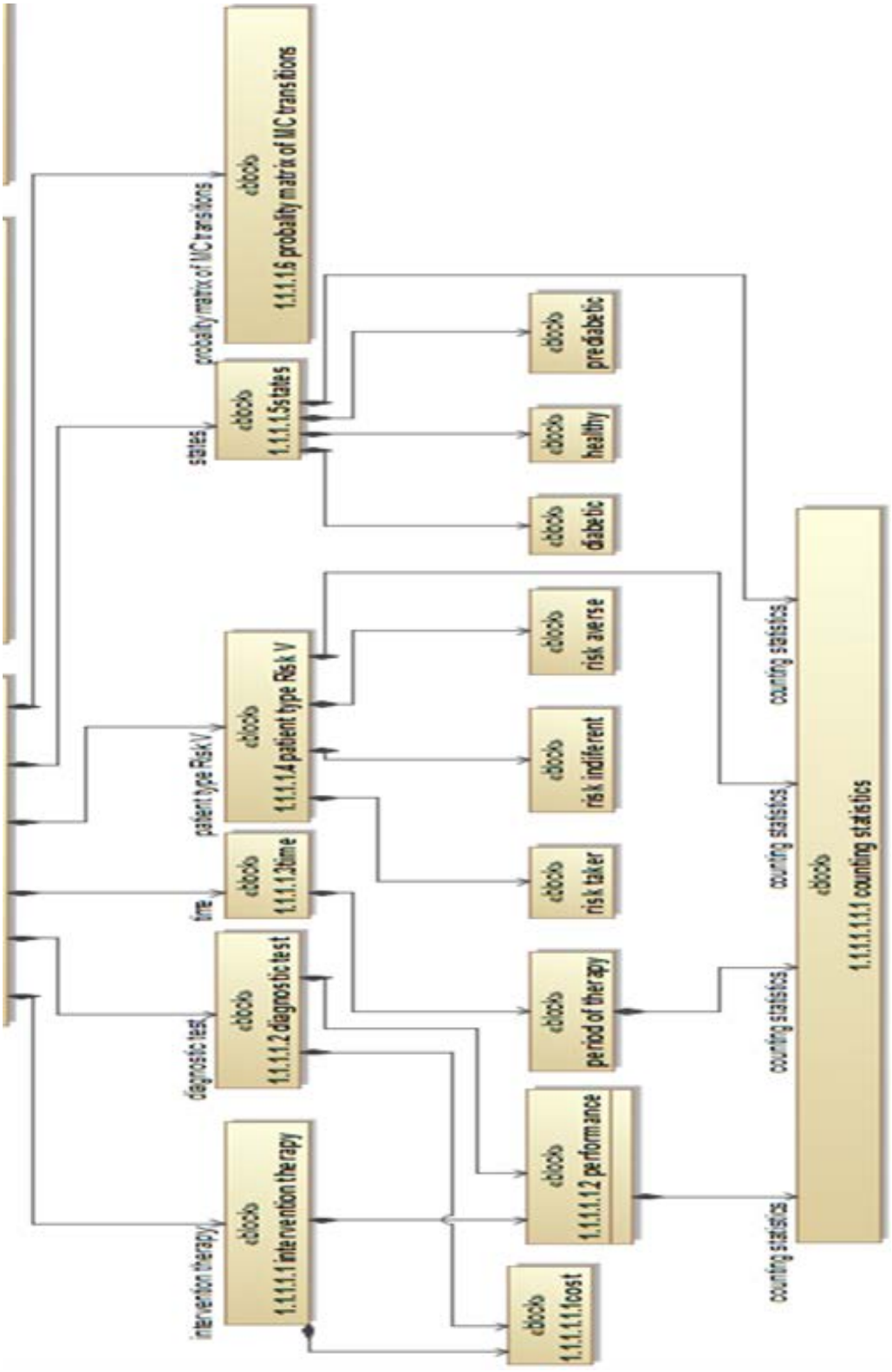


Fig. 27: Structure (Block definition) Diagram for part (the EMCS subsystem) of the Reasoning Engine (MBSE system) of the HCMS for Diabetes Mellitus

Section 8.5: Behavior Diagram of a HCMS for Diabetes Mellitus

As discussed in Chapter 2, a fundamental construct in MBSE is the Behavior (or Block) Diagram [17-20]; which is one of the four pillars of SysML [17-20] and provides a model of the **System Behavior** (i.e. it answers the question "what the system does?"). There are several types of Behavior Diagrams [17-20], dpending on the scale (detail) of the model intended; Sequence Diagrams are used for higher detail while Activity Diagrams are used for more aggregate system behavior modeling. We provide some examples of these fundamental MBSE constructs for components of the Reasoning Engine (MBSE system) of the HCMS for Diabetes Mellitus. In Figure 28 we provide an Activity Diagram (swim-lane type) that describes how the MBSE system works at a very high level (aggregate level) as an interaction of several actors of the system. In Figures 29 and 30, we provide Sequence Diagrams showing the information flow and functionality of two components of the Reasoning Engine (MBSE system) of the HCMS for Diabetes Mellitus: of the EMCS component in Figure 29 and of the FOMCO – SN component in Figure 30.

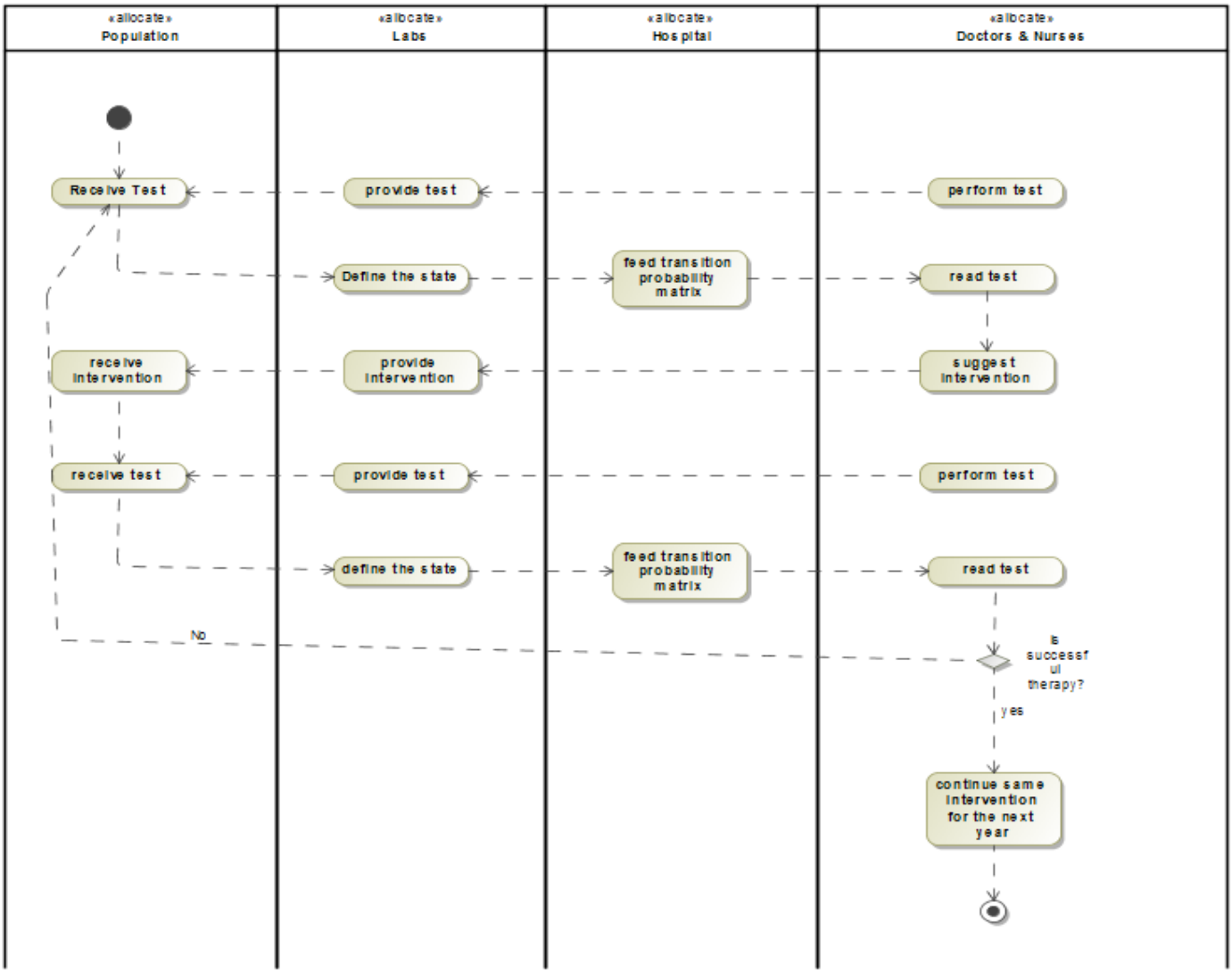


Fig. 28: Activity Diagram of the MBSE system (Reasoning Engine) of the HCMS for Diabetes Mellitus

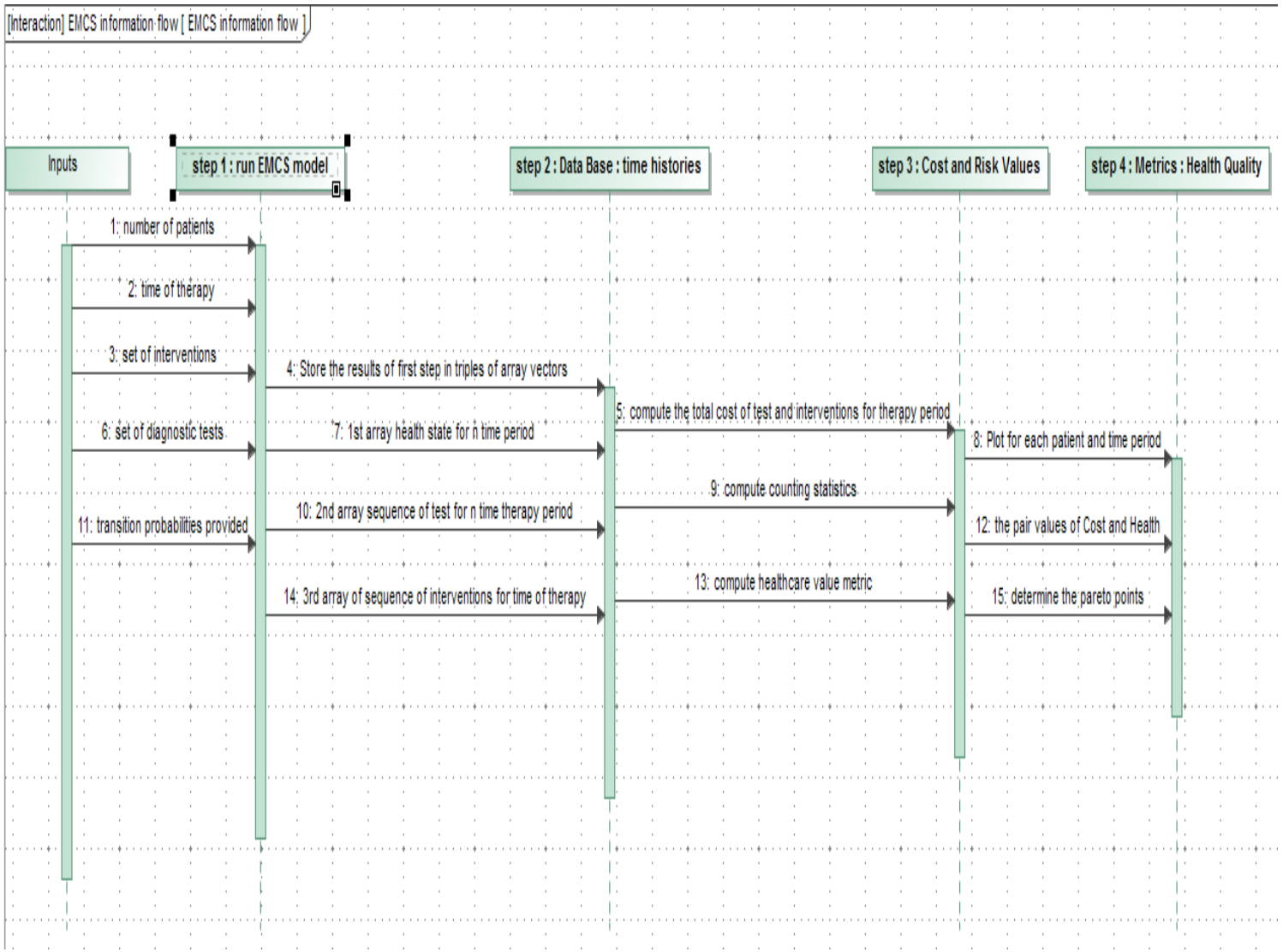

[Interaction] EMCS information flow [EMCS information flow]
Inputs
step 1 : run EMCS model
step 2 : Data Base : time histories
step 3 : Cost and Risk Values
step 4 : Metrics : Health Quality
1: number of patients
2: time of therapy
3: set of interventions
4: Store the results of first step in triples of array vectors
5: compute the total cost of test and interventions for therapy period
6: set of diagnostic tests
7: 1st array health state for n time period
8: Plot for each patient and time period
9: compute counting statistics
10: 2nd array sequence of test for n time therapy period
11: transition probabilities provided
12: the pair values of Cost and Health
13: compute healthcare value metric
14: 3rd array of sequence of interventions for time of therapy
15: determine the pareto points

Fig. 29: Sequence Diagram of the information flow of the EMCS subsystem of the Reasoning Engine of the HCMS for Diabetes Mellitus

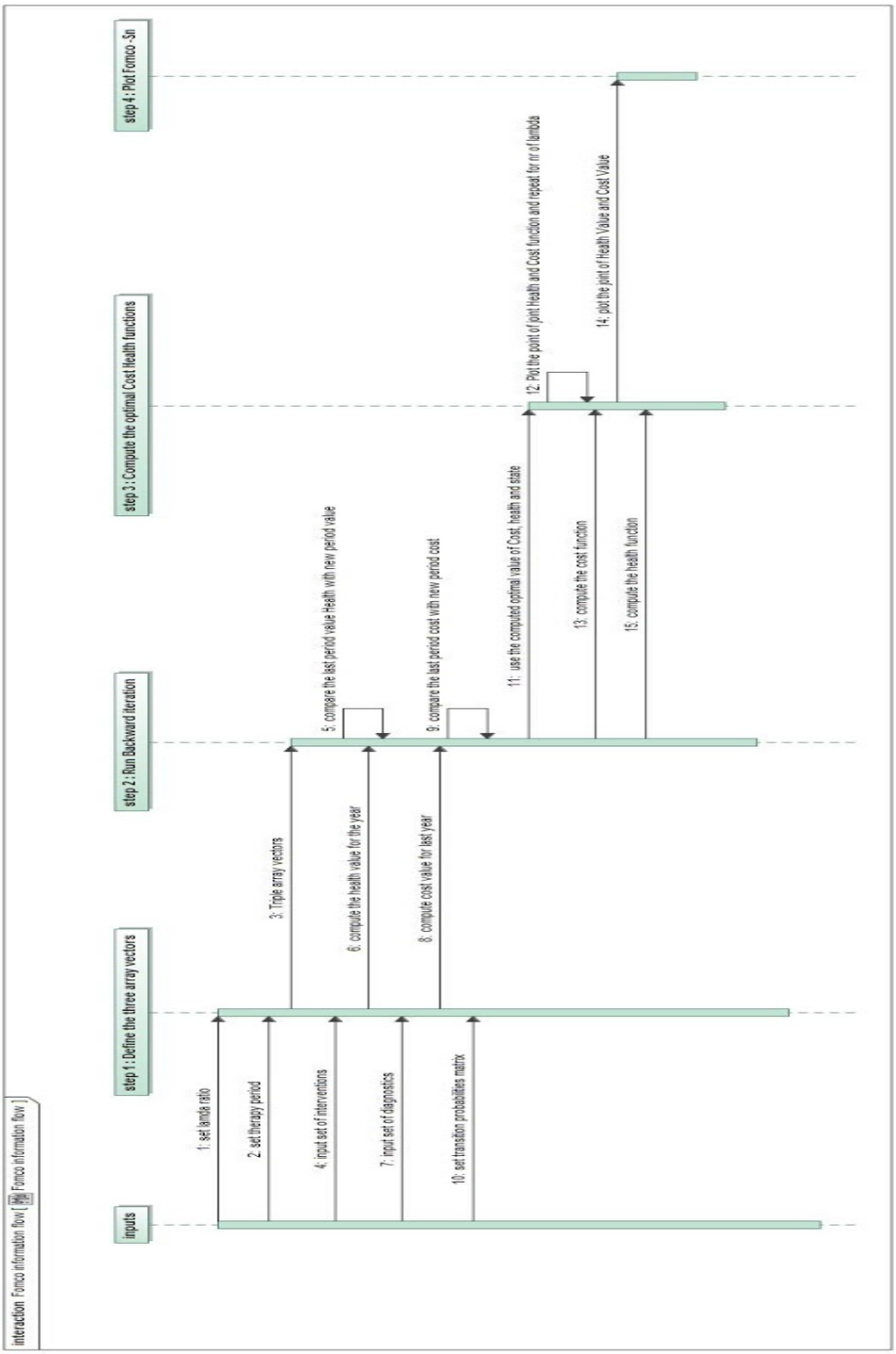


Fig. 30: Sequence Diagram of the information flow of the FOMCO-SN subsystem of the Reasoning Engine of the HCMS for Diabetes Mellitus

Section 8.6: **Requirements Diagram for the EMCS Subsystem of a HCMS for Diabetes Mellitus**

As discussed in Chapter 2, a fundamental construct in MBSE is the Requirements Diagram [17-20]; which is one of the four pillars of SysML, and provides a formal way to capture the requirements for a system, and which can be linked to tradeoff and design space exploration tools [18-20].

In Figures 31 and 32 (placed side by side) we provide a Requirements Diagram for the information flow of the EMCS subsystem of the Reasoning Engine, of the HCMS for Diabetes Mellitus, based on Use Cases and Structure and Behavior Diagrams of EMCS developed (Figures 26 to 30).

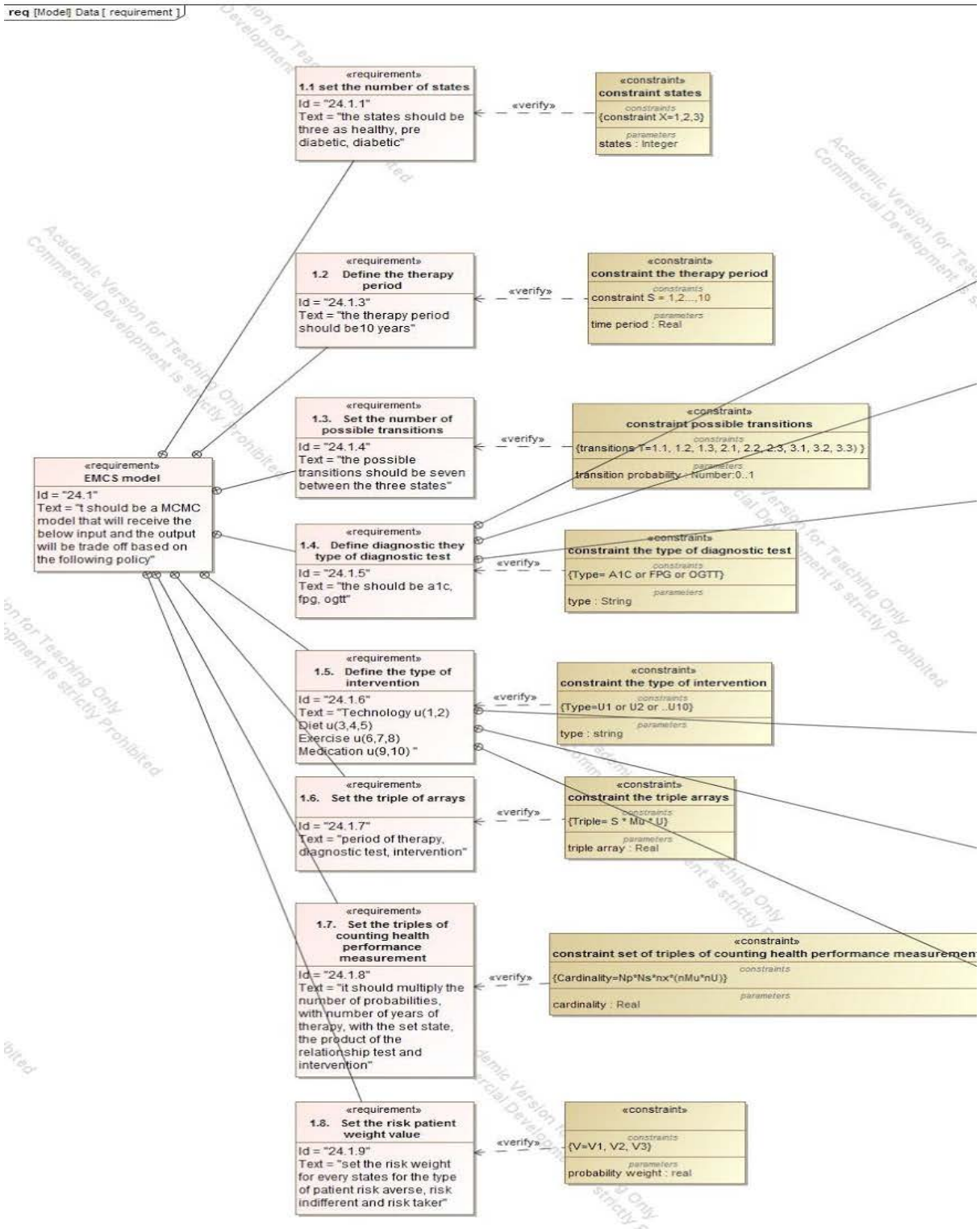


Fig. 31: Requirements Diagram for the information flow of the EMCS subsystem

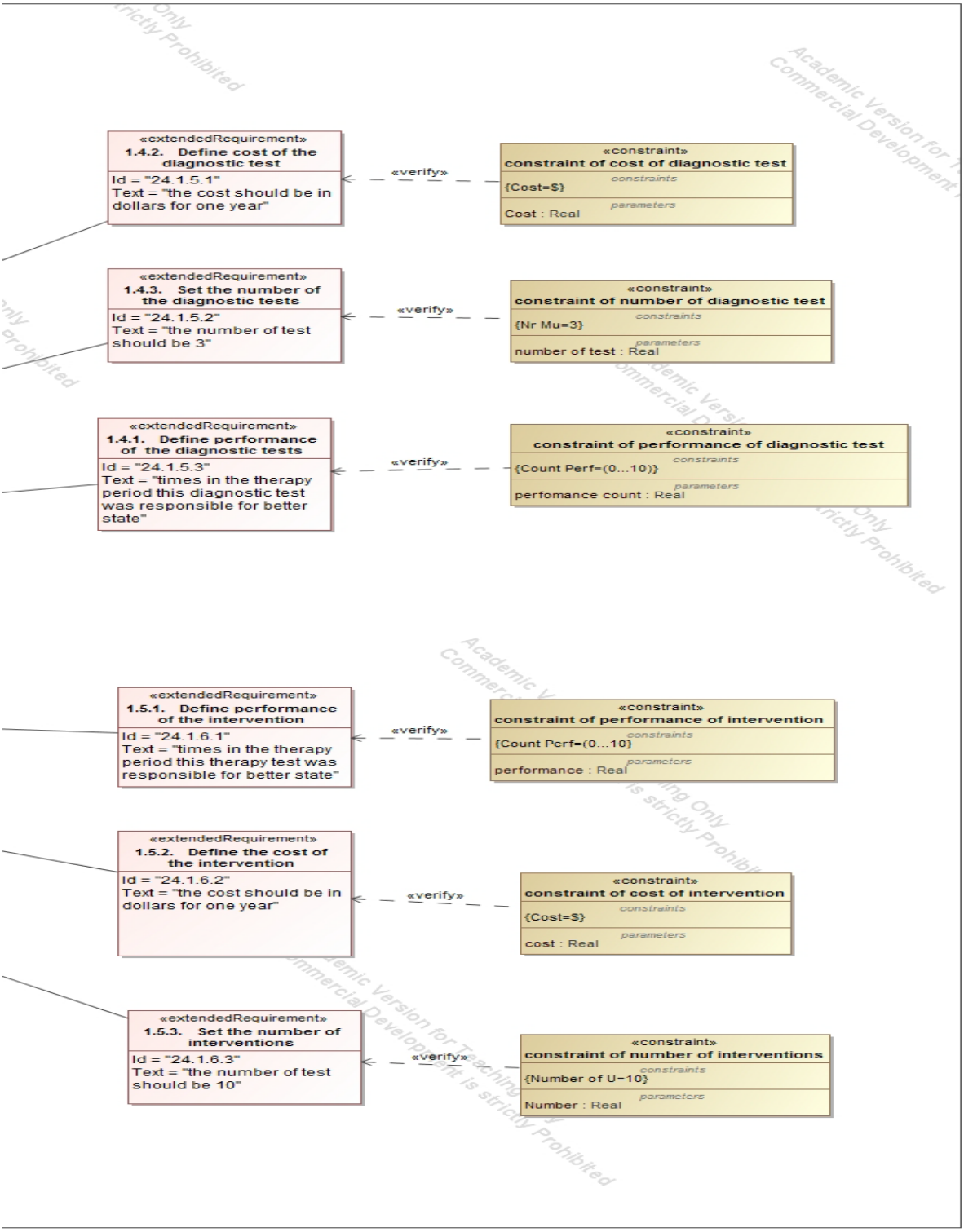


Fig. 32: Requirements Diagram for the information flow of the EMCS subsystem

Section 8.7: Requirements Table for ECMS Subsystem

In the following table, Figure 33, we provide an example of a Requirements Table for the EMCS subsystem of the Reasoning Engine of the HCMS for Diabetes Mellitus. This diagram is also a fundamental construct of MBSE [17-20] and is useful for "book-keeping" requirements indexed by attributes such as id, name, description, constraints, constraint type and parameter. The Requirement id indexing actually provides a hierarchical clustering of requirements corresponding to system components.

Requirement id	Requirement name	Description	Constraint name	Constraint type	Parameter
1	EMCS model	It should be a MCMC that will receive the input and the trade-off based on the following policy			
1.1	Set the number of states	The states should be Healthy, Pre diabetic, Diabetic	states	X=1,2, 3	Integer
1.2	Define the therapy period	The therapy period should be 10 years	therapy period	S=1,..10	Real
1.3	Set the number of possible transitions	The possible transitions should be seven between the three states	Possible transitions	Transitions=1.1…3.3	Pr=0..1
1.4	Define diagnostic	The tests should be	Type of diagnostic	A1c, fpg,	String

	type	a1c, fpg, ogtt	test	ogtt	
1.4.1	Define cost of the diagnostic test	The cost should be in dollars per year	Cost of diagnostic test	Cost=$	Real
1.4.2	Set the number of diagnostic tests	The number of tests should be three	Number of diagnostic tests	Nr Mu=3	Real
1.4.3	Define performanc e of the diagnostic tests	Time in the therapy period this test was responsible for better health state	Performanc e of diagnostic test	Count Perfor mance =3	Real
1.5	Define the type of intervention	Technology (u1, u2), diet (u3..u4), exercise (u6,..u8), medication (u9, u10)	Type of intervention s	Un=U1 ..U10	String
1.5.1	Define performanc e of intervention	Times in the therapy period this intervention was responsible for better state	Performanc e of intervention	Count Perfor mance = 0..10	Real
1.5.2	Define the cost of intervention	The cost should be in dollars per year	Cost of intervention	Cost=$	Real
1.5.3	Set the number of intervention s	The number of tests should be 10	Number of intervention s	Numbe r of U=10	Real
1.6	Set the triples of counting health performanc	It should multiply the number of probabilitie s with the	Triples of counting performanc e measureme	Cardin ality= Np* Ns*Nx *(Nmu	Real

	e measurement	number of the years of therapy, with the set state, the product of pair (intervention, test)	nt	*Nu)	
1.7	Set the risk patient weight value	Set the risk weight for every state for the type of patient: risk averse, risk indifferent and risk taker	risk	V=V1, V2, V3	Pr=0..1 Real

Fig. 33: Requirements Table for the information flow of the EMCS subsystem

Section 8.8: Requirements Traceability Matrix

Requirements Traceability Matrix is another fundamental MBSE construct [17-20], that links requirements and sub-requirements to blocks in the Structure and Behavior Diagrams. Typically the Requirements Traceability Matrix is depicted as a table that links the requirement and/or sub-requirements to the Use Case(s) that generated it, to the Requirement (or Sub-requirement id, and to the Block id in the Structure and Behavior Diagrams [18]. The linkage of use cases, requirements and block definition diagram is the equivalent of the join table in databases and specifically the inner join. Thus we can avoid duplication and we can better extract, transform and load our changes in the database. The

Requirements Traceability Matrix is very useful in design space exploration, as it helps trace satisfaction or non-satisfaction of requirements to specific components and eliminate searches in the design space in non-promising directions. In Figure 34, we show an example of a Requirements Traceability Matrix for the ECMS subsystem based on the Use Cases, Requirements Table, Structure and Behavior Diagrams of the EMCS developed and shown in Figures 26 to 30.

Use case	Requirement	Block Definition Component
1	1.4	1.1.1.1.2
	1.4.1	1.1.1.1.1.1
	1.4.2	1.1.1.1.1.2
	1.4.3	1.1.1.1.1.1.1
2	1.5	1.1.1.1.1
	1.5.1	1.1.1.1.1.1
	1.5.2	1.1.1.1.1.2
	1.5.3	1.1.1.1.1.1.1
3	1.4	1.1.1.1.2
	1.4.1	1.1.1.1.1.1
	1.4.2	1.1.1.1.1.2
	1.4.3	1.1.1.1.1.1.1
	1.1	1.1.1.1.5
	1.3	1.1.1.1.6
4	1.5	1.1.1.1.1
	1.5.1	1.1.1.1.1.1
	1.5.2	1.1.1.1.1.2
	1.5.3	1.1.1.1.1.1.1
	1.1	1.1.1.1.5
	1.3	1.1.1.1.6
	1.2	1.1.1.4
	1.8	1.1.1.3
4	1.1	1.1.1.1.5
	1.3	1.1.1.1.6
	1.2	1.1.1.4
	1.8	1.1.1.3

	1.7	1.1.1.1.1.1
	1.6	1.1.1.1.2

Fig. 34: Requirements Traceability Matrix for the information flow of the EMCS subsystem

Chapter 9: Decision Making and Analytics Capabilities of the Reasoning Engine of a HCMS for Diabetes Mellitus

In this Chapter demonstrate some of the decision making and analytics capabilities that the Reasoning Engine system we constructed, as part of a HCMS for Diabetes Mellitus, enables. ***We would like to emphasize that these capabilities and analytics derive from the MBSE methodology applied***. ***Namely, from the linkage of efficient and powerful tradeoff analysis methods and algorithms for design space exploration, with effective system dynamic models of disease progression that incorporate tests (i.e. measurements, observations), interventions (i.e. actions), and many system and system-environment parameters*** [18, 19]. As emphasized throughout this thesis, our primary contribution is the utilization of modern MBSE methods to develop a ***framework for the design and operation of HCMS for various chronic diseases*** [1, 7]. We selected and used Diabetes Type 2 (Mellitus) [8-13] as a representative example of a chronic disease and as the driving application to motivate and focus our ideas and constructions. By ***framework***, we mean for example the inclusion in such modern and future HCMS of the following key

components: (i) Dynamic models of the behavior of key components of the system; (ii) Linkage of these integrated models with tradeoff analysis tools and design space exploration tools; (iii) Linkage to databases of data, enabling the construction of updated models and action strategies as well as learning; (iv) Incorporation of sensors (diagnostic tests and other type of sensors) and sensory data into the models and searches; (v) Incorporation of actions (controls, interventions) into the models and searches. Thus, we advocate that several instantiations (implementations) of such HCMS are possible, even for Diabetes Mellitus, and we do not advocate the particular one we developed and analyzed in this thesis as the only one. We also advocate that the framework we developed, following MBSE methods and tools, is very promising.

To demonstrate the decision making and analytics capabilities of our system, we pose some interesting and realistic questions from the perspective of health care management and support. These are meant to be examples of problems, questions or queries, the heterogeneous users of the proposed system may want solved or answered. Clearly, many more questions of this type can be posed, and we believe in an implemented system, both the questions and the answers/analysis/interpretations will be archived and made available to the heterogeneous users on a role-based access basis. We generated synthetic data from our models, to answer these questions, provide some rudimentary analysis and justification of each answer, and some interpretation; knowing and understanding very well that these are interpretations based on totally synthetic simulation data, and cannot be perceived as medical statements. Nevertheless

for illustrative purposes they are included in each answer. We generated synthetic data using our disease progression models that incorporate tests, interventions costs etc., various parameters, using the disease models in a Monte Carlo simulation mode. We generated the following data sets:

(a) 100,000 patients, 9 runs, with 3 metrics (Performance, Cost, Reward);

(b) 10,000 patients, 32 runs with 2 metrics (Performance, Cost).

In each data set we include the three types of patients (Risk Averse, Risk Indifferent, Risk Taker) uniformly distributed across types (i.e. the probability for a patient to be of a particular type was 1/3, 1/3, 1/3).

Utilizing our Reasoning Engine and basically the two main tradeoff analysis methods/tools developed (EMCS and FOMCO-SN) we provide in this chapter some examples of analytics and answers to these problems, questions, queries. We organized the work and associated investigations around eight problems/ questions. Many more can be considered and used to exercise the system. Since the data we used are synthetic with somewhat arbitrary parameters (although we chose the parameters carefully form various sources as explained in Chapter 4) these studies are not meant to produce medical evidence, but rather to demonstrate the type of problems that can be solved using our methods, the type of analytics that can be computed, and the associated decision making and interpretations. As we apply both of our methods to the same problem, the results also indicate the differences between the two methods, and the associated advantages and weaknesses of each; these findings point to interesting future research directions and improvements (see Chapter 10). In

the investigations and experiments described below, as well as in the associated graphs, we designate by V1 a Risk Averse patient, by V2 a Risk Indifferent patient and by V3 a Risk Taker patient.

Section 9.1: Effects of Patient Type on Health Care Quality vs Cost: **Based on Comparison of Pareto Frontiers**

We want to investigate the effects of patient type on the main results obtained from our Reasoning Engine (tradeoff analysis based), namely the Pareto points (or Pareto frontiers). Intuitively we expect that patient type will have an effect, and the question is whether our Reasoning Engine can compute results that support our intuition. Once we aggregate all these Pareto points by patient type, we can observe the existence of a relation between Healthcare Quality and Healthcare Cost. For aggregation we can either cluster these Pareto points into sets, or we can fit a curve to these Pareto points, and then check to see whether this curve has the same shape for different risk behaviors of patients, or more generally compare the curves of different patient types. Such differences suggest the existence of an effect of behavioral risk on the performance of health care.

We performed such investigations using both the EMCS and the FOMCO-SN methods. In our experiments and analytics on this problem, we aggregated the totality of Pareto points only on the basis of patient type; that is regardless of patient health state, type of test or intervention used. The only difference was

the behavioral risk of patients (i.e. patient type), which is what we wanted the analytics to focus on.

In Figure 35, we show the results of such an experiment (10,000 patients, 32 runs) and clustering of Pareto points from tradeoffs between two metrics: Cost and Performance. The diagram was derived from the analysis of Pareto data computed by the ECMS method from three categories of patients (V_1, V_2, and V_3), as indicated in the figure. There is a positive trend between the cost and the health performance as expected. It appears that all three patient types have a common area where their costs and performance are in the same range; in Figure 35 this area is confined by the range of values between 0.2 to 0.4 of the Health Value Performance scale, and between 2400 to 2800 of the Total Cost scale. This might be interpreted as a common "health

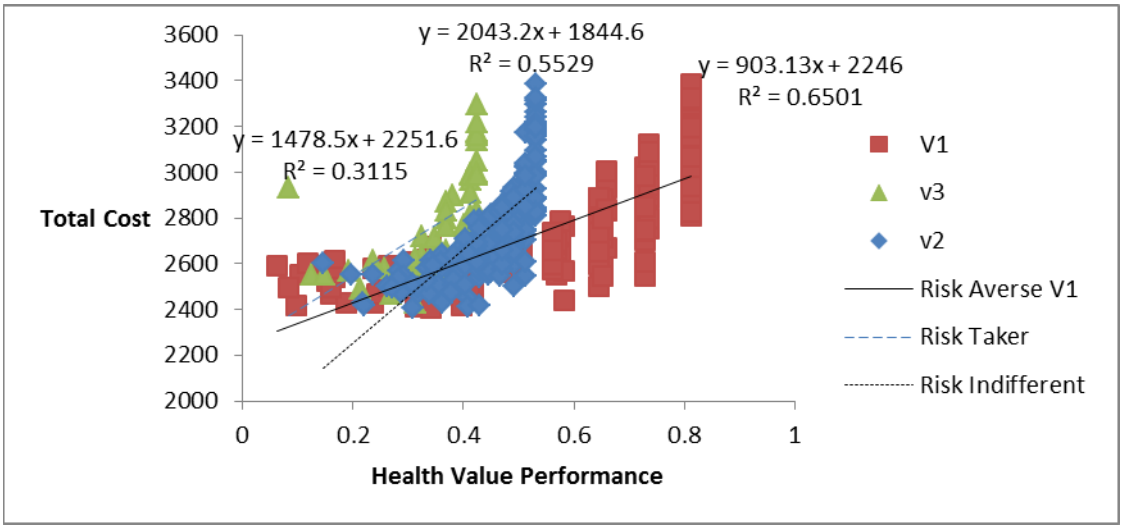


Fig. 35: Pareto points sets (Performance vs Cost) from EMCS, clustered by patient type

maintenance" type pattern. We also observe that all three Pareto frontiers (or Pareto tubes) show an exponential-like increase of cost for increased performance, meaning that all patient types tend to spend disproportionally more to receive little additional healthcare value. Such trends have been also observed in various studies of the increasing healthcare costs world-wide [1-7]. But we also observe that the curves (or more accurately the "tubes" of data) are steeper in the Risk Taker (V_3) and Risk Indifferent (V_2) patients and less steep for the Risk Averse patient (V_1). It seems from the plot that the V_1 patient performs "better" than V_2, and V_3. In other words the Pareto "tube" of a V1 patient lies below the Pareto "tube" of a V2 and of a V3 patient. That is for the same cost, a V1 patient can achieve much higher performance than a V2 and a V3 patient, while for the same performance the cost of V1 is typically lowest and that for V3 highest. With a similar interpretation a V3 patient performs "worse". In a more technical description of the analytics represented in Figure 33, the Pareto frontier (Pareto tube) of a Risk Averse patient (V1) dominates the other two, and the Pareto frontier (Pareto tube) of a Risk Indifferent patient (V2) dominates that of the Risk Taker patient (V2). These interpretations are consistent with expectations based on intuition. The value of a HCMS system, with a Reasoning Engine of the type we propose, is that if supported with real data it could quantify these qualitative expectations, and thus lead to measurably better decisions and policies.

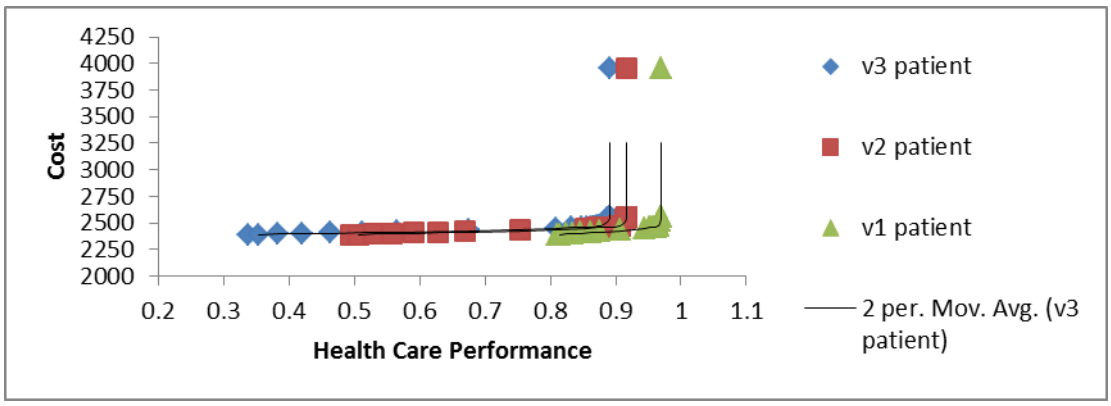


Fig. 36: Pareto points sets (Performance vs Cost) FOMCO-SN, clustered by patient type

In Figure 36, we show results of a similar analysis using the FOMCO-SN method

instead (using 200 sampled values of the scalarization parameter λ, see page 101)

and similar clustering of Pareto points from tradeoff between two metrics: Cost and Performance. The results shown in Figure 34 are from the data set of 100,000 patients (i.e. set (a) page 131). There are similar trends in the graphs and clusters shown in Figures 33 and 34, but there are also important differences. The "best" achievable values for performance are higher for V1, in the middle for V2 and lowest for V3 patients, as expected. There is a positive trend between the cost and the health performance as expected, but its range and variability are much smaller than those found in the analysis reported in Figure 35. Similarly as in Figure 35, we observe that the curves (or more accurately the "tubes" of data) are steeper in the Risk Taker (V_3) and Risk Indifferent (V_2) patients and less steep for the Risk Averse patient (V_1). It seems

from the plots that the V_1 patient performs better than V_2, and V_3. In other words the Pareto "tube" of a V1 patient lies below the Pareto "tube" of a V2 and of a V3 patient. With a similar interpretation a V3 patient performs "worse". However, the tubes of the different type patients are a lot closer in Figure 36 (as compared with Figure 35). These differences (between Figures 35 and 36) are attributed to two causes. First, to the detailed optimization involved in FOMCO-SN; optimizing the selection of treatments at each tie step, optimizing the tests, and continuously striving for the tradeoffs throughout time histories. This eliminates many "bad" Pareto points in the overall set. Second, the averaging involved in FOMCO-SN (while there is no averaging involved in EMCS) eliminates substantially variability. Further research and computational experiments are needed to develop the robustness and accuracy of these promising analytics that are enabled by this type of a Reasoning Engine.

Section 9.2: Effects of Patient Type on Health Care Quality vs Cost: **Subsets of Pareto Points from Distribution Percentiles**

In this Section we investigate further the effects of patient risk type on healthcare cost and performance, from a more statistical perspective. We use the simulations and tradeoff results from our Reasoning Engine, but now we perform some more detailed statistical analysis on the distributions of the values of the metrics we employ. We again cluster the data and Pareto points, by patient type only (patient risk behavior only). But then we look in more detail at these

Pareto points sets, by defining subsets from percentiles of the distribution of values of the metrics we use: Cost, Health Performance, Reward. The motivation comes from the need to see if this type of detailed analysis helps us quantify better the differences in health performance and costs in different patient type classes, than methods that compute averages over the entire sets of data and Pareto points. This analysis can also help understand better, and quantify, which metric is a more important driver of human risk behavior. This type of questions and analysis is common in more sophisticated data analytics, where one wants to look more carefully in the distributions of values of parameters of interest, rather than just their averages.

First we perform the analysis with the EMCS method. We have data from 32 runs on 10,000 patients (i.e. data set (b), see page 131). The details of our computations are as follows. First, we cluster the Pareto points from the whole data set of time histories according to patient type. Let us denote these three sets of 2D points as $P_{V_1}, P_{V_2}, P_{V_3}$. Following the same coordinate axes convention as in Figures 33 and 34, each point in these sets is represented by a two dimensional vector (*HP*, *C*), where *HP* corresponds to the value of Health Performance and *C* to the value of Cost. Then we sort each set of these 2D sets in ascending order with respect to Cost and separately in descending order with respect to Health Performance (i.e. with respect to each one of the coordinates of these sets of 2D points). Let $C_{V_i}^{Top\,20\%}, C_{V_i}^{Bottom\,20\%}$ denote the top 20th percentile and bottom 20th percentile w.r.t. to Costs, in each of the sets $P_{V_1}, P_{V_2}, P_{V_3}$. Similarly let

$HP_{V_i}^{Top\,20\%}, HP_{V_i}^{Bottom\,20\%}$ denote the top 20th percentile and bottom 20th percentile w.r.t. to Health Performance, in each of the sets $P_{V_1}, P_{V_2}, P_{V_3}$. Then we compute the top 20th (lower) Cost percentile subset $P_{V_i}^{LC\,20\%}$ (i.e. points in each of $P_{V_1}, P_{V_2}, P_{V_3}$, with Costs $\leq C_{V_i}^{Top\,20\%}$), and the bottom 20th percentile (higher) Cost subset $P_{V_i}^{HC\,20\%}$ (i.e. points in each of $P_{V_1}, P_{V_2}, P_{V_3}$, with Costs $\geq C_{V_i}^{Bottom\,20\%}$), from each set of Pareto points (one for each patient type). Similarly we compute the top 20th (higher) Health Performance percentile subset $P_{V_i}^{HHP\,20\%}$ (i.e. points in each of $P_{V_1}, P_{V_2}, P_{V_3}$, with Health Performance $\geq HP_{V_i}^{Top\,20\%}$), and the bottom 20th percentile (lower) Health Performance subset $P_{V_i}^{LHP\,20\%}$ (i.e. points in each of $P_{V_1}, P_{V_2}, P_{V_3}$, with Health Performance $\leq HP_{V_i}^{Bottom\,20\%}$), from each set of Pareto points (one for each patient type). Next we compute the average of the Health Performance metric (or index) (i.e. average with respect to the Health Performance coordinate of these sets of 2D points) of the set $P_{V_i}^{HHP\,20\%}$ and the average of the Cost metric (i.e. average with respect to the Cost coordinate of these sets of 2D points) of the set $P_{V_i}^{HHP\,20\%}$ for each of these subsets $P_{V_1}, P_{V_2}, P_{V_3}$ of Pareto points. In Table 5 we show the results of these computations.

There are several observations that can be made. First, these computations reveal more clearly (as compared to Figure 35 and 36) the differences in Health Care Quality (Performance) achievable; i.e. the values of 20th percentiles w.r.t. to Health Performance are very different for the different types of patients. Second,

there is substantially larger difference on the averages of Health Performance from the differences of Costs, between the $P_{V_i}^{HHP\,20\%}$ values and the P_{V_i} values. These and several other observations that can be made support the hypothesis that percentile statistics of subsets of Pareto points reveal better the effects of patient type on Health Performance vs. Cost analysis than averages. One can perform similar analyses with different percentile statistics to reveal better trends and key parametric values of the relationship between patient behavior vs Health Performance vs Cost. Clearly more research and experiments / computations are needed in this direction, especially with real data.

	Risk Averse		**Risk Indifferent**		**Risk Taker**	
	Health Performance	**Cost**	**Health Performance**	**Cost**	Health Performance	**Cost**
Top 20% percentiles Health Perf., Cost	0.736081	2577.8	0.51180556	2918.4	0.311482	2986
Average of Health Perf., Cost, over $P_{V_i}^{HHP\,20\%}$	***0.81251***	2506.111	***0.53248***	2506.634	**0.38201**	2676.925
Average of Health Perf., Cost over P_{V_i}	***0.579465***	2769.3	***0.44885666***	2761.736	**0.086334**	2769.686

Table 5: Comparison of Health Performance vs Costs for subsets of Pareto points clustered by patient type and percentile statistics of Health Performance and/or Cost

statistics of subsets of Pareto points reveal better the effects of patient type on Health Performance vs. Cost analysis than averages. One can perform similar

analyses with different percentile statistics to reveal better trends and key parametric values of the relationship between patient behavior vs Health Performance vs Cost. Clearly more research and experiments / computations are needed in this direction, especially with real data.

We also performed similar studies using the EMCS method again but with three metrics: Cost, Health Performance and Reward (see Section 6.2, page 64). We present the results of one such study in Table 6, where we varied the percentile statistics. We used a data set from 9 runs on 100,000 patients (i.e. data set (a), page 131), with the same model parameters as in the study of Table 5. We run EMCS and compute the Pareto points; each Pareto point is now a 3D point. First, we cluster the Pareto points from the whole data set of time histories according to patient type. Let us denote these three sets of 3D points as $P_{V_1}, P_{V_2}, P_{V_3}$ again. Following the same coordinate axes convention as in Figures 18, 19, 20, each point in these sets is represented by a three dimensional vector (*HP*, *C, R*), where *HP* corresponds to the value of Health Performance, *C* to the value of Cost and R to the value of Reward. Then we sort each set of these 3D sets in ascending order with respect to Cost. Let $C_{V_i}^{Top10\%}, C_{V_i}^{Top20\%}, C_{V_i}^{Top90\%}$ denote the top 10th percentile, 20th percentile, and 90th percentile respectively w.r.t. to Costs, in each of the sets $P_{V_1}, P_{V_2}, P_{V_3}$. Then we compute the top α^{th} (lower) Cost percentile subset $P_{V_i}^{LC\alpha\%}$ (i.e. points in each of $P_{V_1}, P_{V_2}, P_{V_3}$, with Costs $\leq C_{V_i}^{Top\alpha\%}$), where α = 10, 20, 90, from each set of Pareto points (one for each patient type) $P_{V_1}, P_{V_2}, P_{V_3}$. Next we compute

the average of the Health Performance metric (or index) (i.e. average with respect to the Health Performance coordinate of these sets of 3D points), the average of the Cost metric (i.e. average with respect to the Cost coordinate of these sets of 3D points), and the average Reward metric (i.e. average with respect to the Reward coordinate of these sets of 3D points), for each of the subsets $P_{V_i}, P_{V_i}^{LC10\%}, P_{V_i}^{LC20\%}, P_{V_i}^{LC90\%}$ $P_{V_i}^{HHP20\%}$, for i =1, 2, 3, of Pareto points. In Table 6 we show the results of these computations. The first column shows the type of patient. The second column shows the percentile thresholds considered as we mentioned above. The third column shows the average Cost, the fourth column shows the average Reward, and the last column sows the average Health Performance, computed over the subsets of Pareto points defined by these percentiles as discussed above.

Type of patient	Averages	Cost	Reward	H.C P
V1	No Thresholds	2802.042	0.674	0.489
V1	20%	2381.882	1.076	0.722
V1	10%	2294.143	1.134	0.766
V1	90%	2714.318	0.735	0.765
V2	No Threshold	2840.575	0.648	0.490
V2	20%	2416.192	1.028	0.696
V2	10%	2344.154	1.079	0.737
V2	90%	2756.184	0.704	0.523
V3	No threshold	2817.58	0.663	0.485
V3	20%	2405.583	1.043	0.697
V3	10%	2335.583	1.072	0.737
V3	90%	2734.252	0.724	0.512

Table 6: Comparison of Health Performance vs Costs vs Reward for subsets of Pareto points clustered by patient type and percentile statistics of Cost.

Again there are several observations that can be made from these results regarding the effects of patient type. For example V1 type patients get better Health Performance than V2 or V3 type patients, while the corresponding Costs do not differ that much. Similarly for Reward values vs Cost values. As mentioned elsewhere, the point of these examples and computations is not to derive conclusive medically relevant results, but rather to demonstrate the rich set of analytics capabilities that can be built using the Reasoning Engine we have constructed.

Section 9.3: Comparison of Tests and Interventions Based on Health Improvement Impact

A common set of questions that the health care community tries to answer is the relationship between diagnostic tests and interventions, given the time history of patient and/or the time history of the treatment being followed, etc.. Let us show how the Reasoning Engine can be used to compare tests and interventions. Some typical questions and reasoning follow. Which test for each type of patient has the best impact on the patient time history? How one can use the data to try to evaluate this type of a question? For example one can select from the data the time histories where there was a reversal of the disease progression; that is the patient became diabetic at some time, in the horizon we use, and then became healthy at the end of the horizon. One can rank this subset of time histories according to the time steps it took to reverse the disease; the smaller number it took to improve the health care represented in this time series. Can one identify

the tests that were used in the best three subsets of such time histories? We can of course repeat this entire set of analysis for interventions. We can then also compare the results of such studies for each patient type, and study whether such a comparison reveals anything interesting.

Typical medical informatics practice collects data on many similar patients and/or patient time histories, which include results following the application of a test and/or an intervention in terms of disease state transitions, time durations the patient's health stays at a particular state, health care costs, and other data and measurements of health care results. One example of such healthcare studies includes studies that are based on a comparison of a mainstream therapy (test and intervention sequences) with a novel and pioneering one. As discussed already in several parts of this thesis another strong factor in the success or not of a therapy is the behavior of the patient; the more cooperative the patient is with the doctor, the higher the probability that the intervention would be successful.

In this Section we describe some representative analytics that can be performed using the Reasoning Engine we constructed. We describe studies that we performed using the EMCS method, where we analyzed patient time histories with every combination of tests and interventions and we compare the test and interventions employed based on their efficiency and risk patient type. The Monte Carlo generated results of time histories with tests and interventions sequences, show time histories (trajectories) through different health states. We

examined the state trajectories looking for transitions from unhealthy to healthy states. ***These improving health transitions are considered successes of the particular test and intervention used***. These transitions, that lead from unhealthy to healthy, are captured for each test and/or intervention and are compared to the total number that this particular test and/or intervention was used. The ratio between the number of successful uses of a test/intervention and their total number of uses is the efficiency ratio for this test and intervention and for this particular patient time history.

We performed many studies of this type on synthetic simulation data. In Table 7 we show the results of a representative such analytics that address some of the questions described in the beginning of this section. In Table 7:

- The first column shows the method applied to compute the Pareto points and thus determine the interventions to use in each time period;
- The second column shows the specific intervention by giving its number (based on the index of their database);
- The second column shows the time periods (from the ten in each time history, summed over all time histories used) at which the specific intervention was used (from the Pareto points computed);
- The third column shows the time periods (from all time histories) when the use of the intervention led to an improvement in the patient health state;

- The fourth column shows the efficiency of each intervention (the ratio of the number of improvement times vs the total number the intervention was used).

Model and Method used	Patient type	Interventions	Number of times implemented specific intervention	Number of improvement times	Efficiency
EMCS (2 variables)	V1	6	711	129	18%
EMCS (2 variables)	V2	8	26	6	23%
EMCS (2 variables)	V3	8	28	6	21%
EMCS (3 variables)	V1	7	53	19	36%
EMCS (3 variables)	V2	8	36	16	44%
EMCS (3 variables)	V3	8	38	17	45%
FOMCO	V1	6	n/a	n/a	53%
FOMCO	V2	6	n/a	n/a	75%
FOMCO	V3	6	n/a	n/a	75%

Table 7: Best performed intervention for V1, V2, and V3 risk type of patients.

From inspection of the results in Table 7, one can conclude that intervention 6 was

more efficient for patient type V1 (Risk Averse), while intervention 8 was most efficient for patients type V2 (Risk Indifferent) and V3 (Risk Taker).

Section 9.4: Comparison of Tests and Interventions Based on Specific Health Improvement State Transitions

This Section is very similar in concept with the Section 9.3, but with finer detail. Namely we investigated the relationship between tests and interventions that leads to health improvement, but we also clustered (grouped) these improvement transitions based on which of the three health states this improvement happened. In a similar fashion as in Section 9.3, in this Section we analyze every combination of tests and interventions used in all time histories in our data, grouped them by specific health state improvement and then we compare them by their efficiency and risk patient type. The Monte Carlo simulations generated results of tests and interventions that lead to different state trajectories (time histories of patient health states). We used the EMCS method to compute Pareto points, and thus the tests and interventions to be applied at each time instant. The health state transitions from diabetic to pre-diabetic, from pre-diabetic to healthy and from diabetic to healthy were captured, the tests and interventions involved were identified, and then we compared the number of applications resulting to health improvement transitions to the total number that this test and intervention were used. The ratio between the number of successful tests/intervention applications and the number of all applications is as before the efficiency of the test and intervention.

We performed many such studies in the synthetic data we generated. In Tables 8 and 9 we show the results of representative such analytics; Table 8 shows results on evaluation of interventions, while Table 9 shows results on evaluation of diagnostic tests. In Tables 8 and 9:

- The first column shows the patient type;

- The second column shows the type of state transition (from sick to healthier);
- The third column shows the intervention causing the health improvement;
- The fourth column shows the total number of times this specific intervention was used over all patient time histories, for each patient type;
- The fifth column is the total number of times the specific intervention caused the specific improvement state transition;
- The sixth column shows the efficiency of the intervention (which is the fraction of the numbers in the fifth and fourth columns).

Type of patient	Improvement between states	Type of intervention	Times intervention was used in the therapy	Times of improvement	Efficiency
Risk averse	2 to 1	8	28	1	4%
Risk averse	3 to 2	6	680	34	5%
Risk averse	3 to 1	7	131	4	3%
Risk indifferent	2 to 1	8	26	1	4%
Risk indifferent	3 to 2	6	672	30	4%
Risk indifferent	3 to 2	7	128	5	4%
Risk indifferent	3 to 1	6	672	16	2%
Risk indifferent	3 to 1	7	128	3	2%
Risk taker	3 to 2	6	680	34	5%
Risk taker	2 to 1	8	28	1	4%
Risk taker	3 to 1	6	680	34	5%

Table 8: Evaluation of intervention for specific improved health state transitions, for each patient type.

Type of patient	Improvement between states a to b	Type of test	Times of intervention used in the therapy	Times of improvement	Efficiency
Risk averse	2 to 1	2	769	80	10%
Risk averse	3 to 2	1	741	35	5%
Risk averse	3 to 1	3	497	19	4%
Risk indifferent	2 to 1	1	690	89	13%
Risk indifferent	3 to 2	3	503	21	4%
Risk indifferent	3 to 1	1	690	12	2%
Risk indifferent	3 to 1	3	503	9	2%
Risk taker	2 to 1	1	689	87	13%
Risk taker	3 to 2	2	681	27	4%
Risk taker	3 to 2	3	493	22	4%
Risk taker	3 to 1	1	689	14	2%
Risk taker	3 to 1	2	681	13	2%
Risk taker	3 to 1	3	493	10	2%

Table 9: Evaluation of diagnostic test for specific improved health state transitions, for each patient type.

We can also collect all the results for "best" health improvement state transitions caused by specific interventions, for each patient type, and for each possible health improvement state transitions. Table 10 shows the results that can be used the relationship between specific interventions, patient type and specific next state health improvement. The nomenclature for each column is similar as before and is provide in the headings of the columns of Table 10.

Type of patient	Health state improvement	Intervention	Number of times applied	Number of Improvement Times	Efficiency
V1	3 to 2	1	54	16	30%
V1	2 to 1	3	193	27	14%
V1	3 to 1	3	193	27	14%
V2	3 to 2	1	81	21	26%
V2	2 to 1	3	251	55	22%
V2	2 to 1	8	36	8	22%
V2	3 to 1	3	251	55	22%
V2	3 to 1	8	36	8	22%
V3	3 to 2	1	72	20	28%
V3	2 to 1	3	237	40	17%
V3	3 to 1	3	237	40	17%

Table 10: Best performing interventions and their efficiencies, for health state improvement from every state and patient type

Section 9.5: Effects of Patient Type on Health Quality vs Cost: Based on Pareto Frontiers with New Health Quality Metric

This section is related with Section 9.1. Here we take as Health Quality metric the number of time periods (i.e the number of years out of ten years) a patient is in the Healthy state (i.e. Diabetes disease state 1). This Health Quality metric is simply the counting statistic $O_1^i(m_i)$ (see (15), Chapter 5 , page 60). This type of study shows a different analytics capability of our Reasoning Engine. Namely the ability to use different metrics for tradeoff analysis and additional analytics as the ones we have discussed and demonstrated in this Chapter. This is an important capability in order to be able to analyze the robustness of decision making and analytics in HCMS, with respect to the analytic model used for a metric (or metrics). Previously (Section 9.1) we analyzed, how the Pareto frontier was affected by the patient type (health risk behavior) for the Health Quality metric defined in Chapter 5. Here, we investigate a similar question regarding Health Quality and Cost tradeoff, when we consider the new Health Quality metric. We are interested to investigate whether the Pareto front changes and by how much. Such study may also reveal behavioral trends regarding the value patients place on their health vs their type.

We used the EMCS method to compute Pareto points, selected the Pareto points, that are related with Health state one and we grouped them by the behavioral risk of patient (by patient type). Then we plot the new Pareto frontier together with the Pareto frontier that we have in Section 9.1 and check for differences.

In Figures 37 and 38 we show representative results of such computations for Risk Averse and Risk Taker patient types.

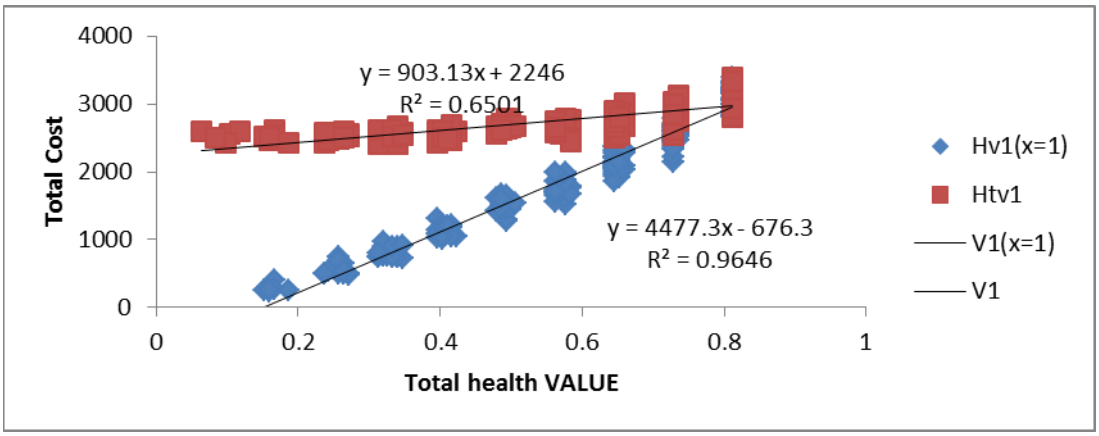


Fig. 37: Total cost to Health Value between all states and keeping the Healthy state only for Risk Averse patients (V1).

In the graph of Figure 37, we have comparable representation of the normal health performance vs cost curve including all the health states. The second curve, which follows a linear behavior between cost and health value, is the health performance value when only the state healthy state was selected.

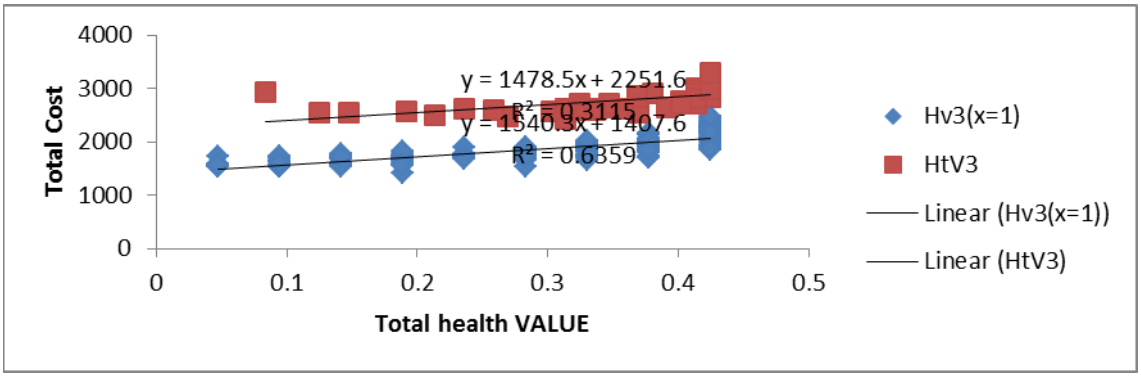


Fig. 38: Total cost to Health Value between all states and keeping the Healthy state only for Risk Taker patients (V2).

Chapter 10: Conclusions and Directions for Future Research

The present thesis investigated the value of System Engineering, and especially of Model Based System Engineering (MBSE), as a basis for a framework for the development of Health Care Management Systems (HCMS) for chronic diseases. We used Diabetes Mellitus as a driving specific case. We developed such a framework and demonstrated the value of a MBSE approach. We focused on the development and evaluation of the Reasoning Engine component of a HCMS. We showed that tradeoff analysis methods linked with integrated models of disease progress that incorporate diagnostic tests and interventions provide a powerful foundation for such a Reasoning Engine, as it allows the investigation of a rich variety of "what-if" type questions.

Following modern MBSE, we developed Use Case, Structure, Behavior, Requirements Diagrams and Requirements Traceability Matrix for components of the Reasoning Engine. We developed a Controlled Markov Chain model for the progression of Diabetes Mellitus, with three states, three diagnostic tests and ten interventions. We developed several metrics in our new framework and focused on two: Health Care Quality metric and Cost metric. We considered three types of patient behavior: Risk Averse, Risk Indifferent and Risk Taker. We also showed how to generate several other metrics including a Reward metric. We developed two basic methods for tradeoff analysis and developed, implemented

and tested the associated algorithms, that were incorporated in the Reasoning engine of the HCMS. The first, **Evaluation by Monte Carlo Simulation (EMCS)** generates all possible patient time histories with all possible test and intervention sequences, and then computes Pareto points by direct comparison. The second, **Fully Observable Multi-criteria Optimization (FOMCO)** is based on the use of Dynamic Programming (DP) for multi-criteria optimization. Two cases of FOMCO were developed. The first, FOMCO-SN, is based on scalarization and normalization, and combines the metrics for trade off analysis in one via convex combination. The second FOMCO-E, selects one metric to optimize and treats the remaining ones as constraints. FOMCO-SN is more efficient numerically than FOMCO-E and it was selected for further development, implementation and experimentation. Based on our optimization results for Pareto points for the EMCS method and Stochastic Dynamic Programing for FOMCO-SN, we conducted a rich analysis, based on several trade-off "what if" scenarios, as "imitations" of real world decision making and analytics in the healthcare management environment. We compared the results of both methods and we discovered some computational advantages of one method over the other. We demonstrated the decision making and analytics capabilities of the Reasoning engine we developed by investigating several practical and important problems for healthcare management and support.

The limitation that we faced during this thesis research was the lack of similar studies in System Engineering, Health Operations and Health Technology Assessment. The basic methods that we used are used in Operational Research

and Control Theory, and there have been a few health care applications, but not in Diabetes Mellitus management. Other limitations that we observe during this research were the luck of datasets for health performance, or cost of tests and interventions. Clinical studies tend to focus mostly in comparison analysis of non-therapy and therapy, but they don not use a common cost function or a common health value metric. Most of the clinical studies are based on close datasets and the validation of their results and conclusions is limited. Therefore we generated simulated data sets for our studies and analytics, by assuming values for costs and other model parameters consistent with the literature on Diabetes Mellitus. Last but not least, there is a clear need for collaboration between physicians, and engineers for resolving this problem in a realistic manner.

Future work will be the continuation of the model development and better linkage to clinical and medical databases and incorporation of machine learning methods. Another direction we will pursue is the formulation, development, simulation, analysis and evaluation of the **Partially Observed Multi-criteria Optimization (POMCO)** method. The Partial Observed Multi-criteria Optimization could be used as decision making tool for a physician or health care provider for Diabetes Mellitus, based on their selection of solutions we can map, and attract their decision making rationalizing and come up with decision trees for analysis. The decision trees could also be compared with machine learning techniques like the one of Random Forest, trying to cluster the actions of a physician. Furthermore, as part of future research should be close cooperation

with physicians, hospital or Accountable Care Organizations for implementation of the various resulting techniques, algorithms and visualization of trade-offs explorations of what-if scenarios. Finally, the present framework needs to be deployed as software, with abilities to extract information, transform the results based on the queries and optimization methods and load the results in meaningful visualizations for more accurate and advanced decision making.

A key output form our Reasoning Engine of a HCMS, are the Pareto points (or Pareto frontiers) that describe succinctly the relative value of treatments and tests vs the overall health care quality of a patient's time history. Running ECMC with two metrics (and 2-D graphs) for 10,000 patients and 32 runs, took for the whole experiment 783sec. Running ECMS with three metrics (and 3-D graphs) for 100,000 patients took for the whole experiment 1,385 sec. FOMCO outputs directly the Pareto-points and other related information and is very fast. For the same problems that the First Method (EMCS) took 783 sec (two metrics, 10,000 patients, 32 runs) and 1,385 sec (3 metrics, 100,000 patients, 9 runs), our Second Method (FOMCO-SN) took only 2.36 sec on the same laptop. Preliminary experiments with initial POMCO implementations took only twice the time of FOMCO. The Pareto points and frontiers computed are very similar to those computed with EMCS but at two or more orders faster time as shown in the tables just below!

Profile Summary

Generated 20-Dec-2014 22:20:23 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
first_method_reward	1	783.910 s	4.289 s	
xlswrite	2080	728.682 s	2.718 s	
actxserver	2080	440.175 s	440.000 s	
xlswrite>ExecuteWrite	2080	284.417 s	69.104 s	
iofun\private\openExcelWorkbook	2080	145.377 s	137.871 s	

Profile Summary

Generated 20-Dec-2014 20:59:38 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
main1_new	1	1384.996 s	110.346 s	
xlswrite	2548	867.597 s	8.982 s	
actxserver	2548	518.934 s	518.799 s	
monteCarlo	3000000	392.592 s	392.592 s	
xlswrite>ExecuteWrite	2548	337.844 s	81.990 s	
iofun\private\openExcelWorkbook	2548	174.645 s	166.385 s	
onCleanup>onCleanup.delete	3088	73.997 s	0.398 s	
xlswrite>@0xlsCleanup(Excel.file)	2548	73.579 s	0.145 s	
iofun\private\xlsCleanup	2548	73.434 s	73.131 s	

Profile Summary

Generated 20-Dec-2014 20:31:58 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
main	1	2.009 s	0.573 s	
monteCarlo	10000	1.336 s	1.336 s	

Method Name	Function Name	Computing Time
ECMS (Cost, Performance)	Main	1,384.996 sec
	Monte Carlo (random variables generator)	867.597 sec
FOMCO- SN	Main	2.009 sec
	Monte Carlo(random variables generator)	1.336 sec

Table 11: Computational Time comparison between EMCS and FOMCO-SN methods

Appendix 1: **2D Graphs from 32 Runs on 10,000 Patients**

In Appendix 1, we include a whole set of 2D graphs for 32 simulation runs of our model for 10,000 patients. These 2D graphs produced by our MBSE system, provide the Pareto frontier (red points) (Health Performance vs Cost) for each patient history. All three types of patients were included.

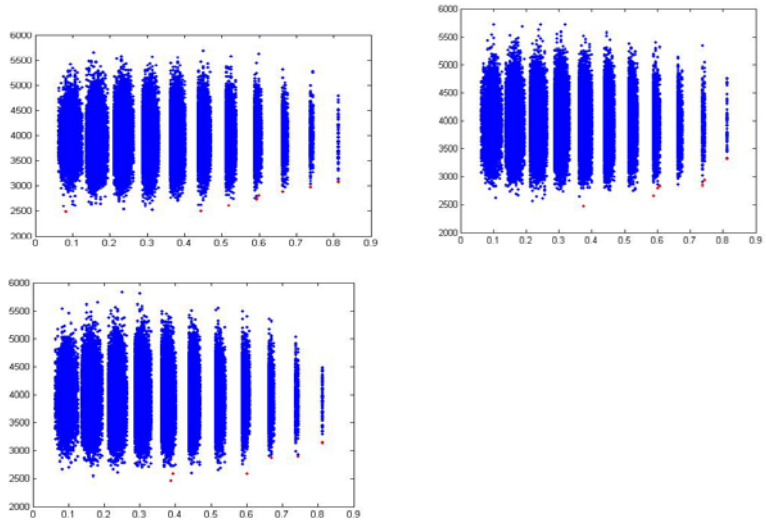


Figure 1 sample 1-3 patient V1

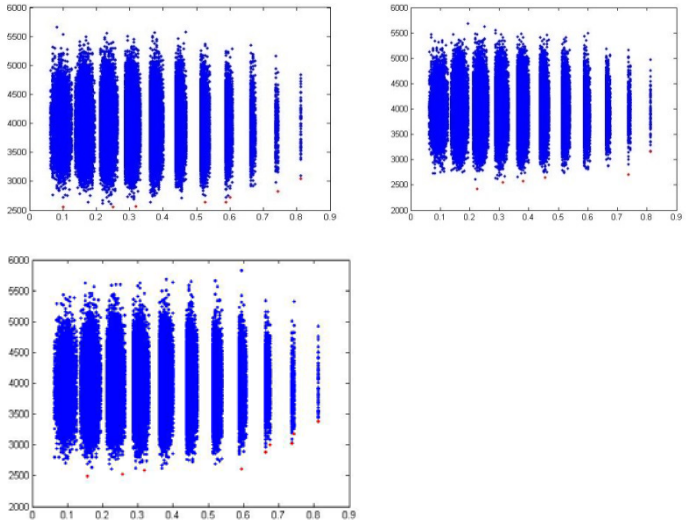


Figure 2 sample 3-6 patient V1

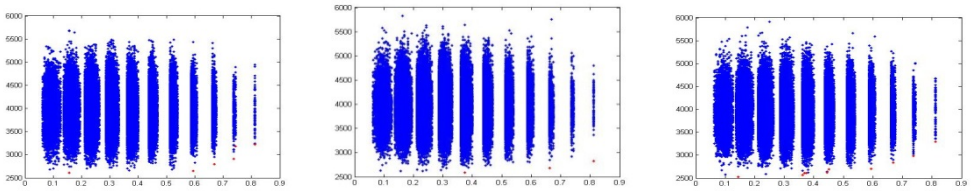


Figure 3 sample 6-9 patient V1

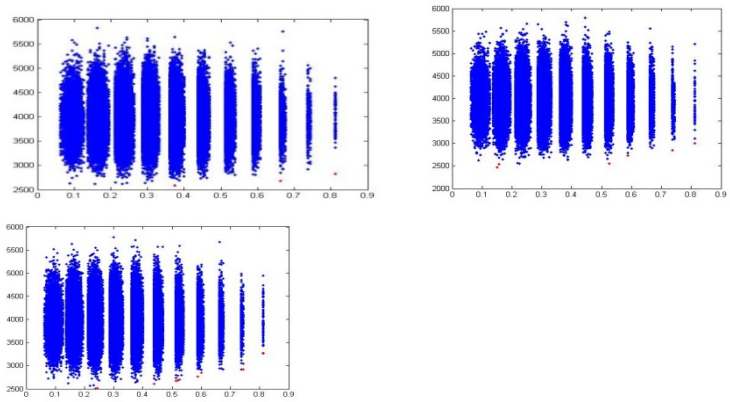


Figure 4 sample 9-12 patient V1

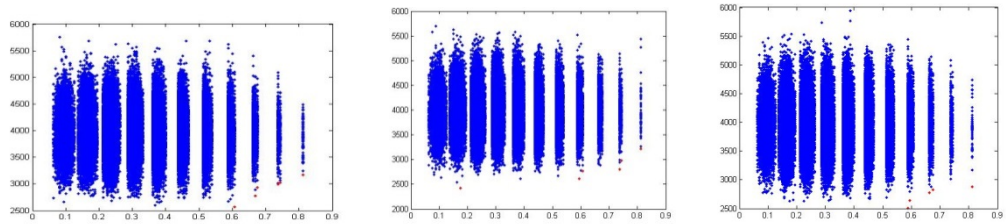


Figure 5 sample 12-15 patient V1

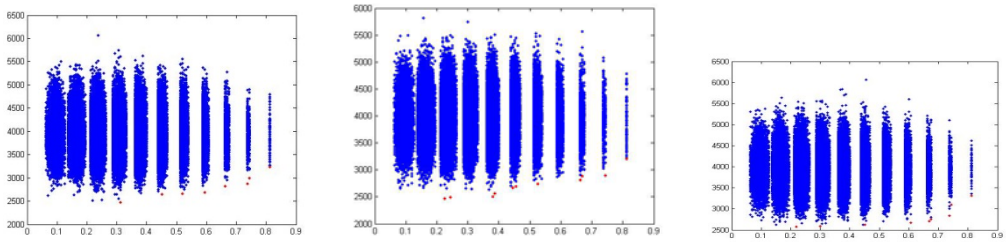


Figure 6 sample 15-18 patient V1

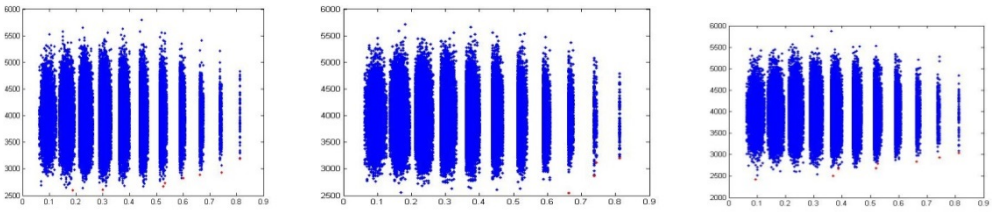


Figure 7 sample 18-21 patient V1

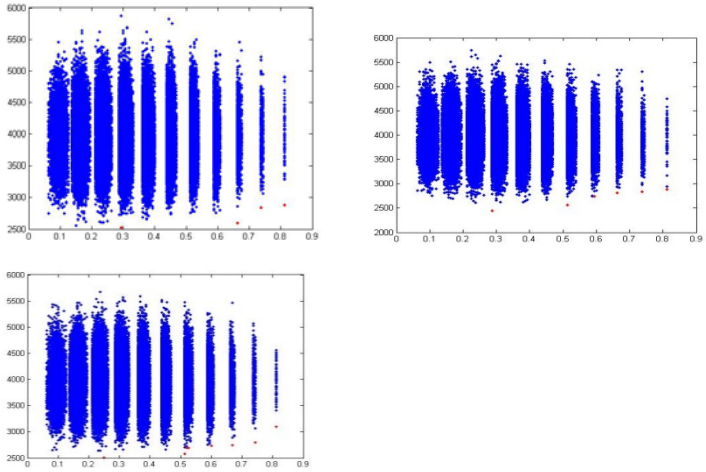


Figure 8 sample 21-24 patient V1

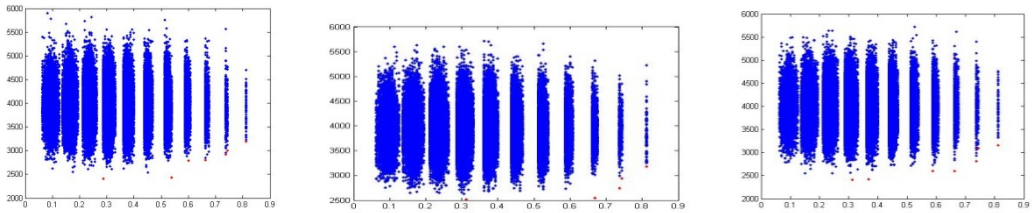


Figure 9 sample 24-27 patient V1

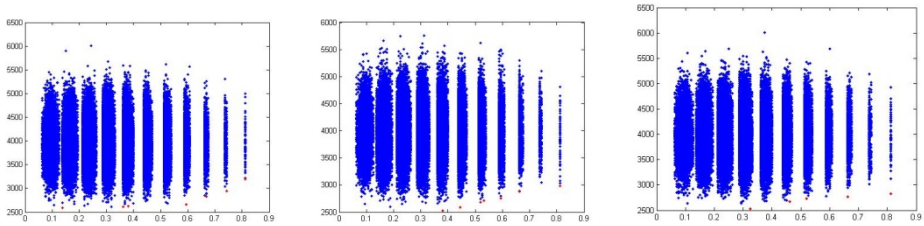


Figure 10 sample 27-30 patient V1

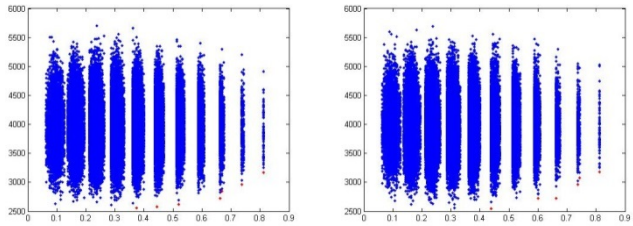


Figure 11 sample 31-32 patient V1

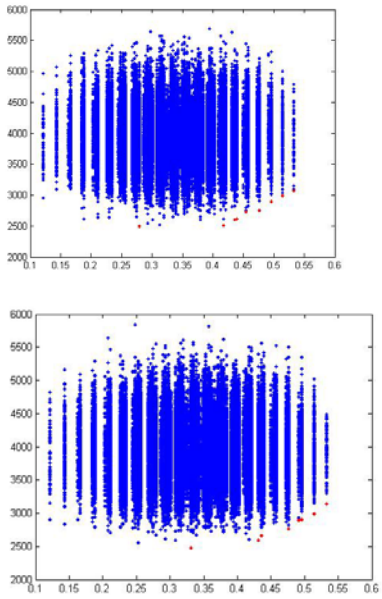


Figure 12 sample 1-3 patient V2

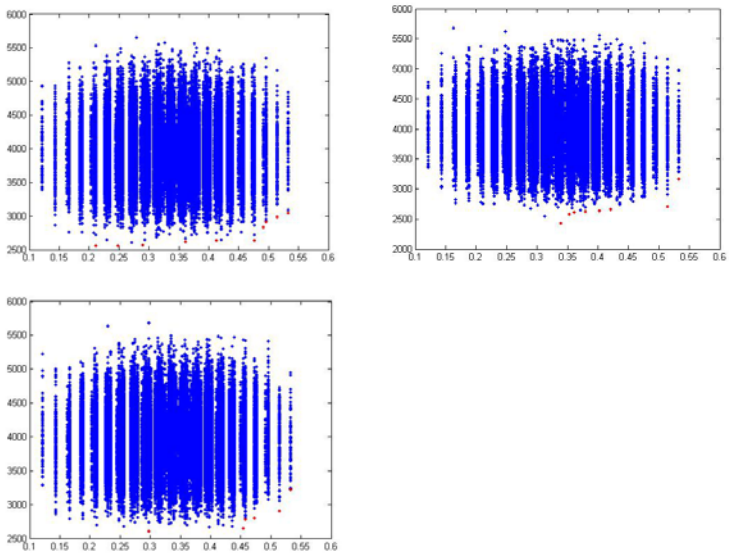


Figure 13 sample 3-6 patient V2

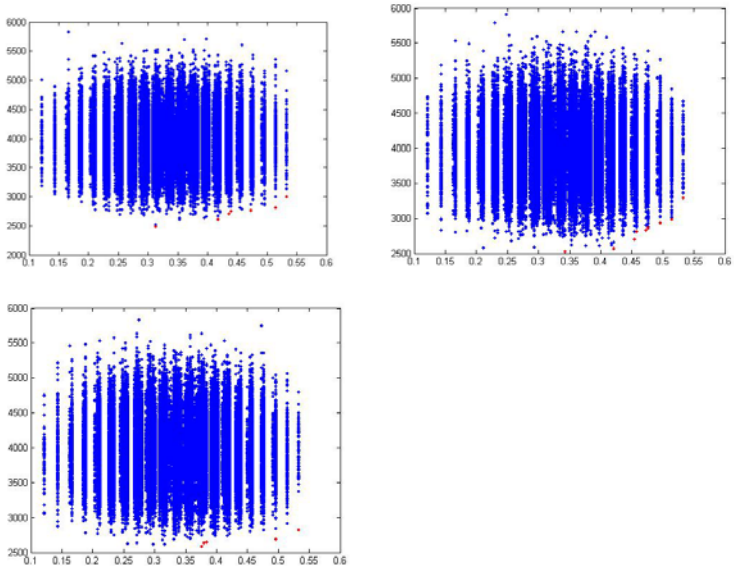


Figure 14 sample 3-6 patient V2

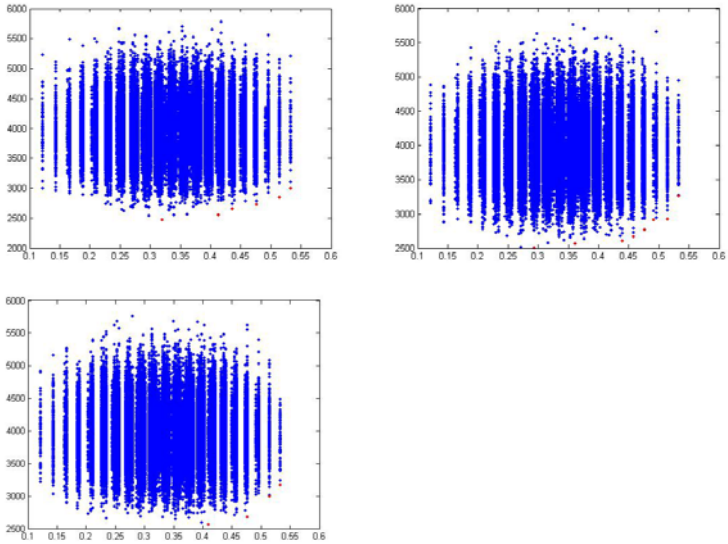


Figure 15 sample 6-9 patient V2

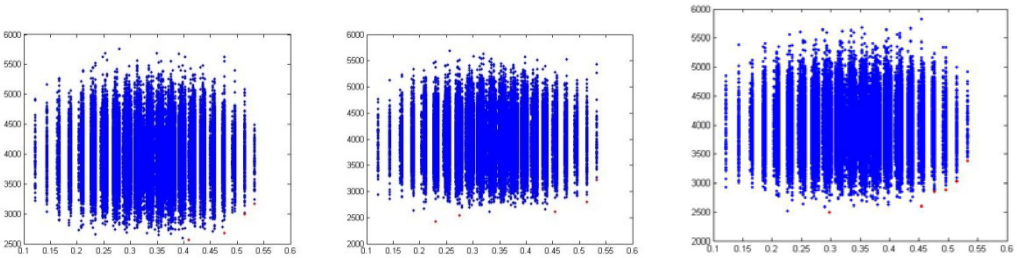


Figure 16 sample 12-15 patient V2

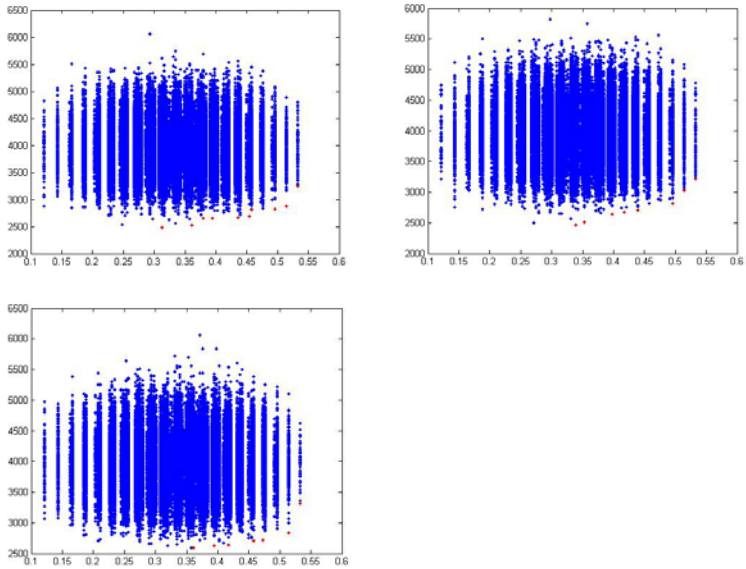


Figure 17 sample 15-18 patient V2

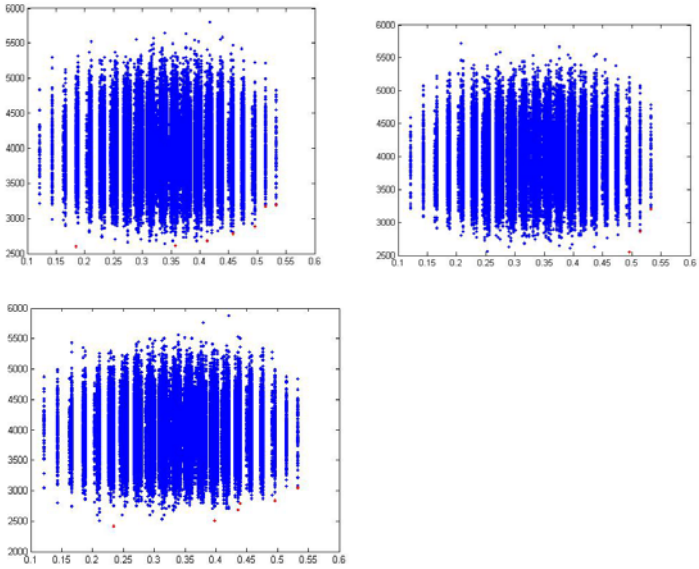


Figure 18 sample 18-21 patient V2

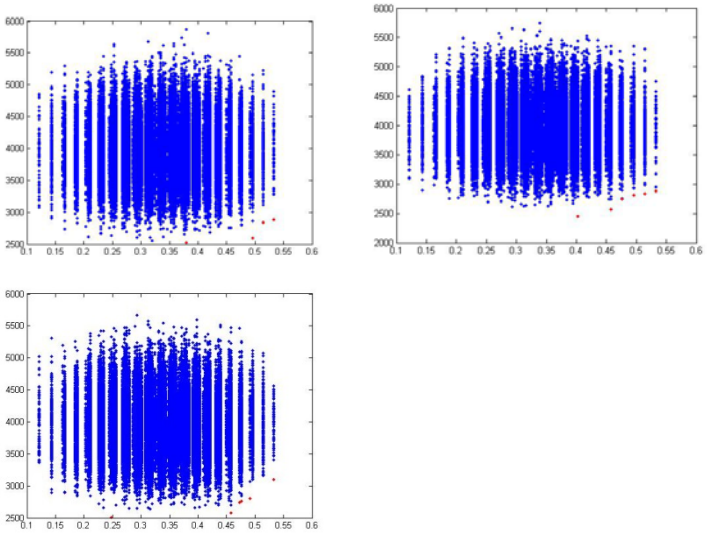


Figure 19 sample 21-24 patient V2

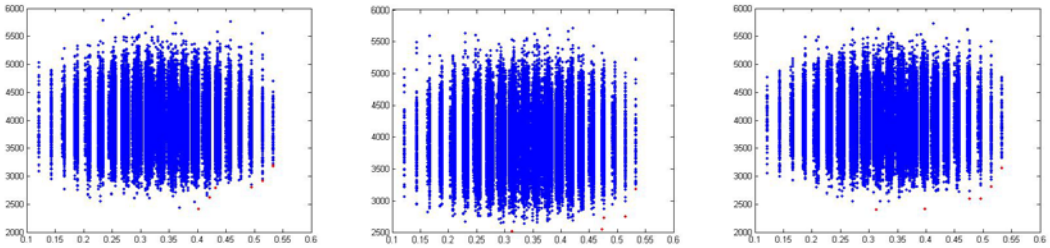


Figure 20 sample 24-27 patient V2

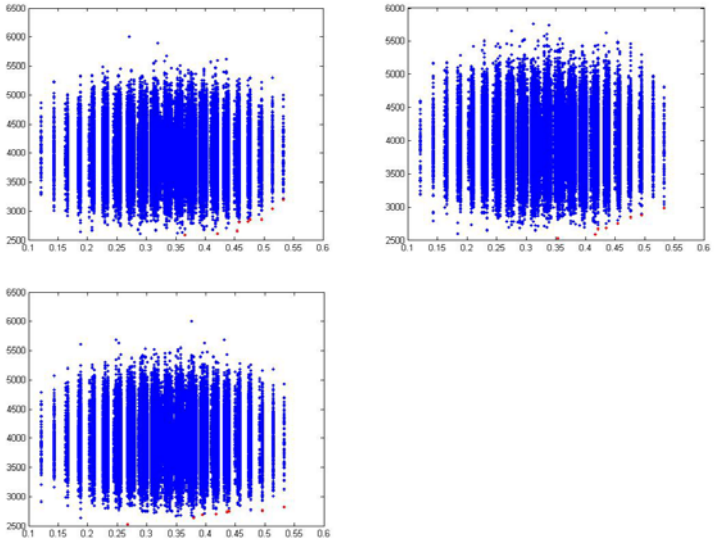


Figure 21 sample 37-30 patient V2

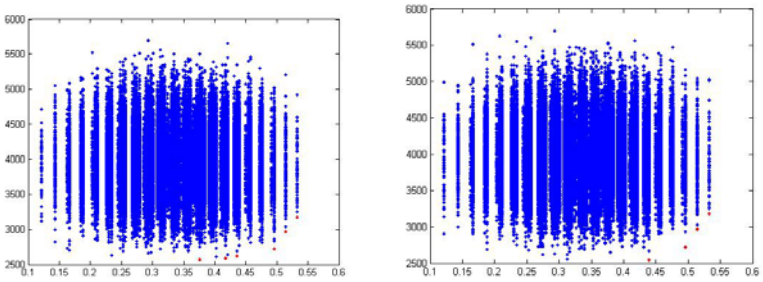


Figure 22 sample 31-32 patient V2

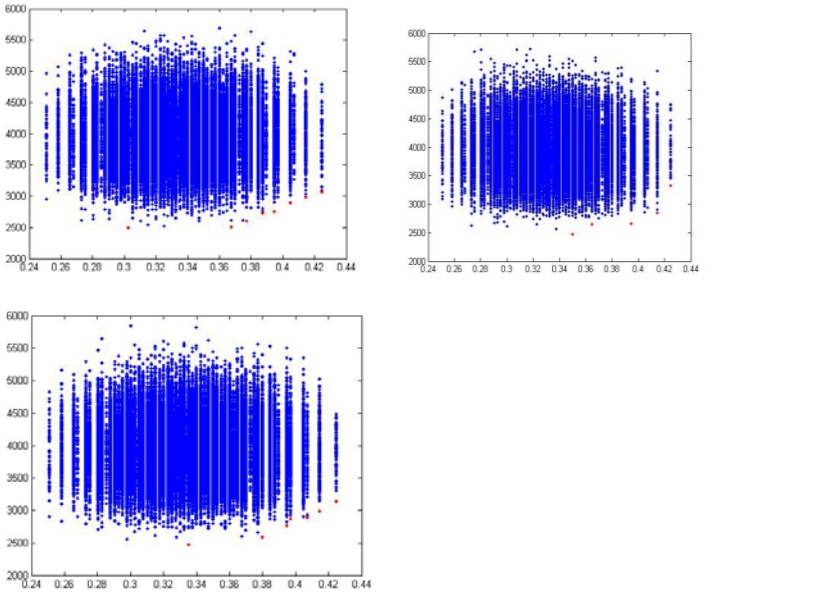


Figure 23 sample 1-3 patient V3

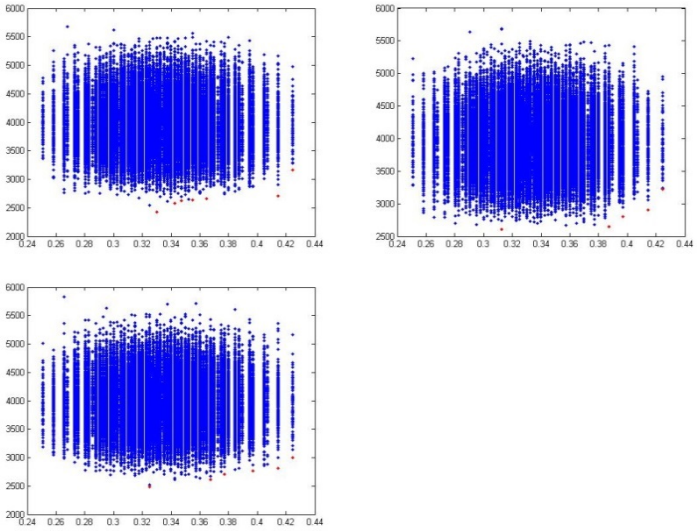


Figure 24 sample 3-6 patient V3

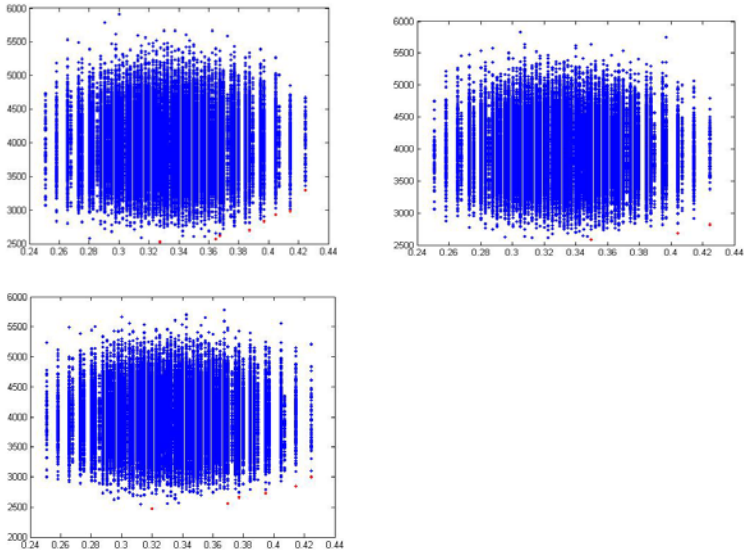


Figure 25 sample 6-9 patient V3

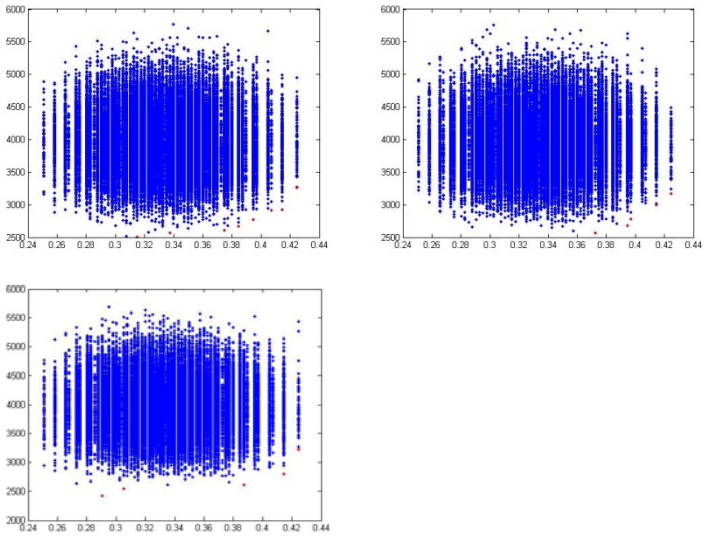


Figure 26 sample 9-12 patient V3

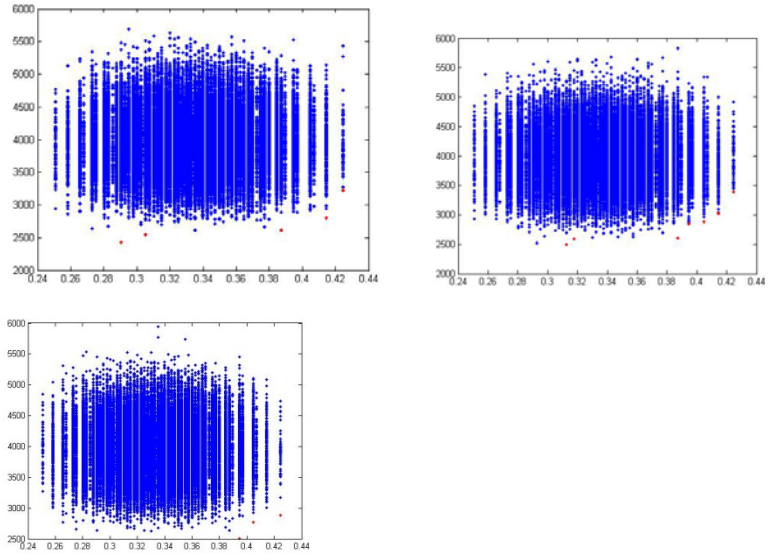


Figure 27 sample 12-15 patient V3

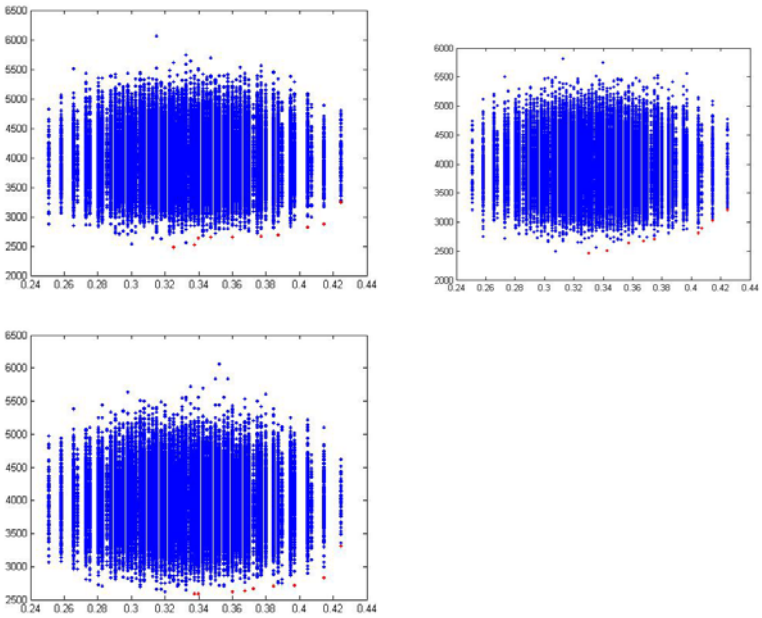


Figure 28 sample 15-18 patient V3

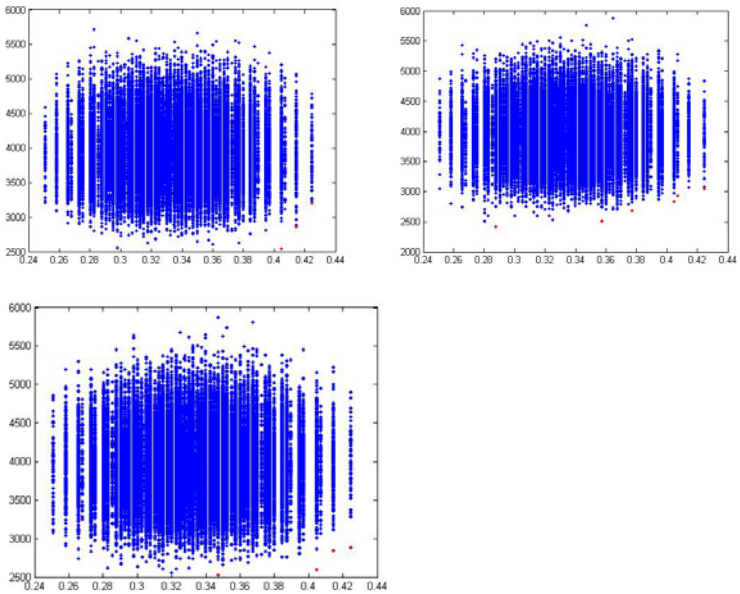


Figure 29 sample 18-21 patient V3

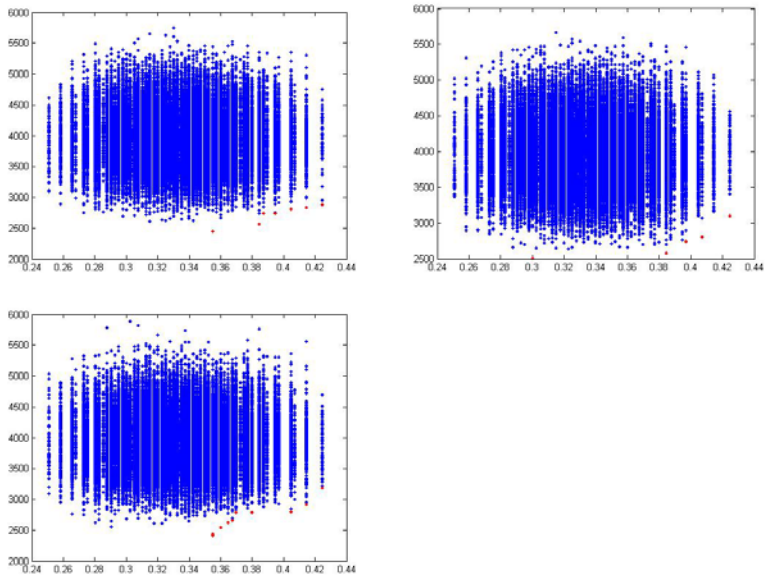


Figure 30 sample 21-24 patient V3

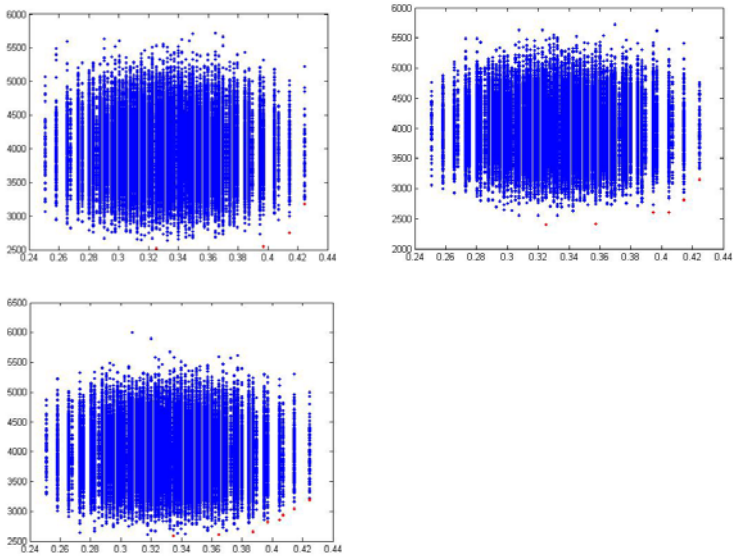


Figure 31 sample 24-27 patient V3

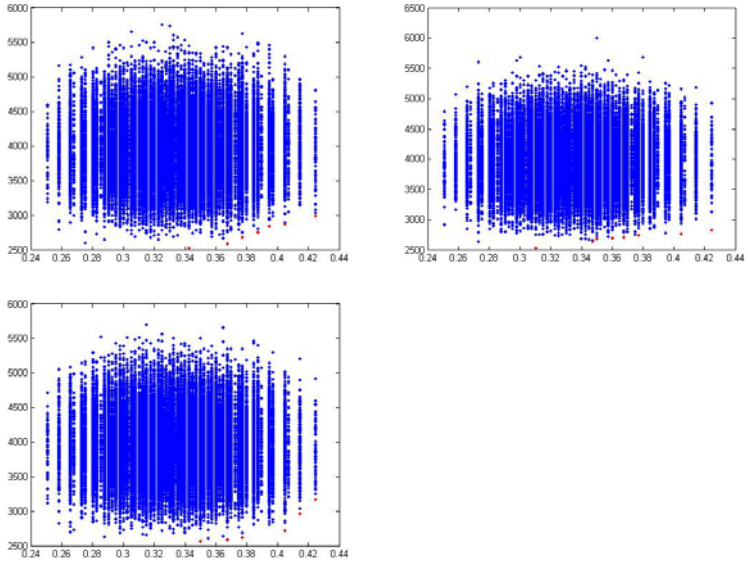


Figure 32 sample 27-30 patient V3

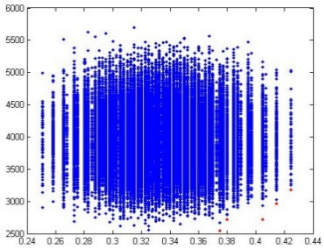
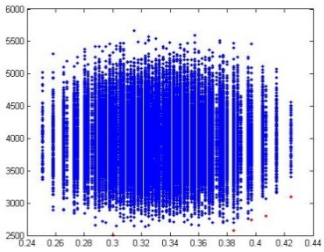


Figure 33 sample 31-32 patient V3



Appendix 2: **3D Graphs from 9 Runs on 100,000 Patients**

In Appendix 2, we include a whole set of 3D graphs for 9 simulation runs of our model for 100,000 patients. These 3D graphs produced by our MBSE system, provide the Pareto frontier (red points) (Health Performance vs Cost vs Reward) for each patient history. All three types of patients were included.

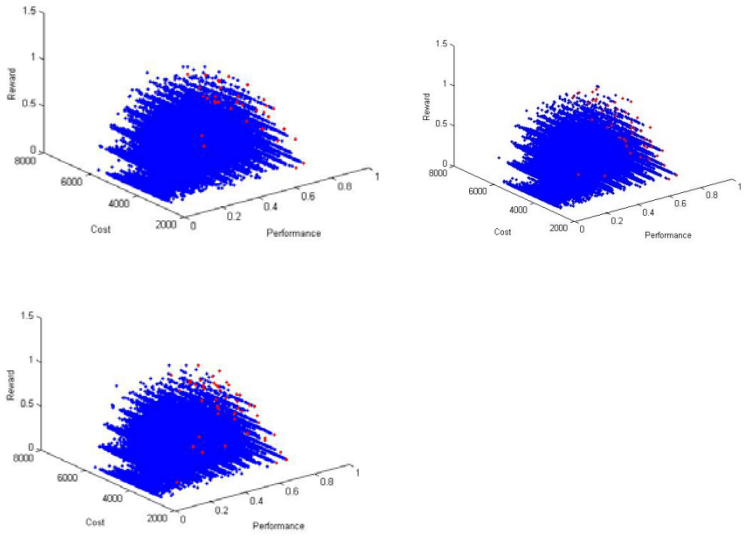


Figure 34 V1 patient runs for 1-3 for 100k patients

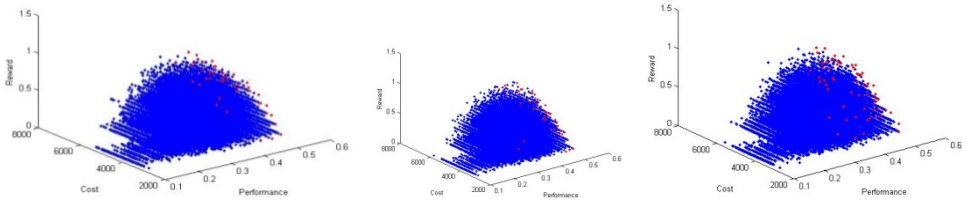


Figure 35 V2 patient runs for 1-3 for 100k patients

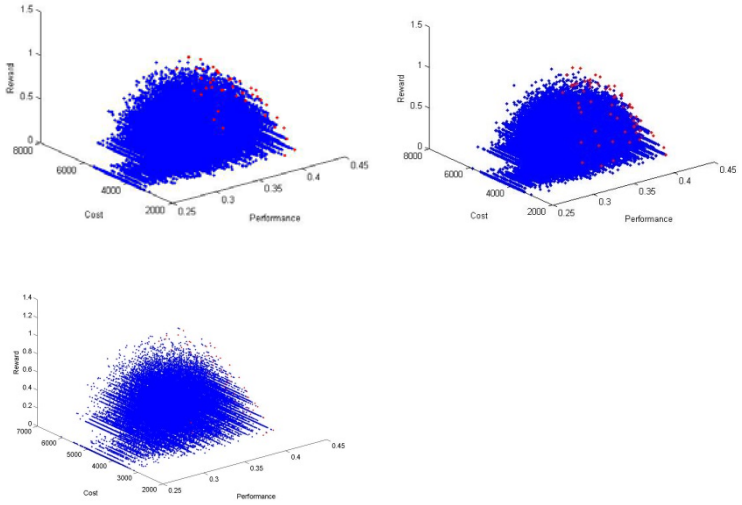


Figure 36 v3 patient runs 1-3 for 100k patients

.

Bibliography

1. C. Cassel et al, "Better Health care and Lower Costs: Accelerating Improvement Through Systems Engineering," *Report to the President, Executive Office of the President*, President's Council of Advisors on Science and Technology, May 2014.

2. Deloitte, "2014 Global Health care Outlook: Shared Challenges, Shared Opportunities," *Deloitte Development LLC, Report*, 2014.

3. A. Shah, "Health Care Around the World," Global Issues, Article 774, September 2011, http://www.globalissues.org/article/774/health-care-around-the-world.

4. Z. Jakab, "Facing Future Challenges for Global Health," *World Health Organization (WHO) R&D Presentation*, London, Nov. 2012.

5. Intrahealth, "Top 10 Global Health Issues to Watch in 2014," *Intrahealth Organization Report*, January 2014.

6. KPMG Global Healthcare, "Staying Power, Success Stories in Global Healthcare," *KPMG International, Report*, 2014.

7. *KPMG, Healthcare,* http://www.kpmg.com/global/en/industry/healthcare/pages/default.aspx.

8. R. Colagiuri et al, "Global Diabetes Plan 2011-2021", International Diabetes Federation (IDF), Brussels, Belgium, 2011.

9. "National Diabetes Statistics Report: Estimates of Diabetes and its Burden in the United States, 2014," *Report*, *U.S. Department of Health and Human Services*, National Center for Chronic Disease Prevention and Health Promotion, Division of Diabetes Translation, Atlanta, GA, 2014. http://www.cdc.gov/diabetes.

10. "A Snapshot of Diabetes in the United States," *Centers for Disease Control and Prevention*, Division of Diabetes Translation, 2014. http://www.cdc.gov/diabetes.

11. "A Snapshot of Prediabetes in the United States," *Centers for Disease Control and Prevention*, Division of Diabetes Translation, 2014. http://www.cdc.gov/diabetes.

12. W. C. Knowler, E. Barrett-Conner, S. E. Fowler, et al, Diabetes Prevention Program Research Group, "Reduction in the Incidence of Type 2 Diabetes with Lifestyle Intervention or Metformin," *N. Engl. J. Med.* 2002;346:393–403.

13. J. Tuomilehto, J. Lindstom, J. Eriksson, et al; Finnish Diabetes Prevention Study Group, "Prevention of Type 2 Diabetes Mellitus by Changes in Lifestyle Among Subjects With Impaired Glucose Tolerance," *N. Engl. J. Med*. 2001; 344:1343–1350.

14. S. Y. Kwankam, "Successful Partnerships for International Collaboration in e-Health: The Need for Organized National Infrastructures," *Bull. World Health Organ.*, 90, 395-397, 2012. doi:10.2471/BLT.12.103770.

15. D. McMorrow et al, "A Robust Health data Infrastructure," *Technical Report JSR-13-700*, JASON, The Mitre Corporation, Nov. 2013.

16. "The Massachusetts Health Information Highway (the HIway), Fact Sheet," 2012. www.mass.gov/hhs/masshiway.

17. International Council on Systems Engineering (INCOSE), *Systems Engineering Vision 2020*, Version 2.03, TP-2004-004-02, 2007.

18. J.S. Baras, *Lecture Notes for MSSE class*, ENSE 621, 2002.

19. D. Spyropoulos and J. S. Baras, "Extending Design Capabilities of SysML with Trade-off Analysis: Electrical Microgrid Case Study," *Proceedings of the Conference on Systems Engineering Research (CSER'13)*, pp. 108-117, Atlanta, GA, March 19-22, 2013.

20. J. S. Baras and M. A. Austin, "Development of a Framework fr CPS Open Standrads and Platforms," *ISR Technical Report 2014-02*, Institute for Systems Research, University of Maryland, November 8, 2013.

21. K. Deb and K. Miettinen, *Multiobjective Optimization*, Springer 2008.

22. *Standards of Medical Care in Diabetes – 2014*, American Diabetes Association, Diabetes Care, Vol. 37, Supplement 1, pp. S14-S80, Jan. 2014.

23. *The Essential Diabetes Book*, Second Edition, Mayo Clinic, 2012.

24. *Insulin Resistance and Prediabetes*, NIH Publication No. 14-4893, National Institute of Diabetes and Digestive and Kidney Diseases, June 2014.

25. *Diagnosis of Diabetes and Prediabetes*, NIH Publication No. 14-4642, National Institute of Diabetes and Digestive and Kidney Diseases, June 2014.

26. Mayo Clinic Diabetes Educational Resource Center, http://diabetesresourcecenter.mayoclinicproceedings.org/.

27. M. I. Schmidt, B. B. Duncan, H. Bang, J. S. Pankow, C. M. Ballantyne, et al, Atherosclerosis Risk in Communities Investigators, "Identifying Individuals at High Risk for Diabetes: The Atherosclerosis Risk in Communities Study," *Diabetes Care*, 28(8):2013-8, Aug 2005. doi:10.2337/DIACARE.28.8.2013

28. K. M. Unerti, M. B. Weinger, et al, "Describing and Modeling Workflow and Information Flow in Chronic Disease Care." *J. Am. Med. Inform. Assoc*, 2009 Nov-Dec, 16(6):826-36. doi: 10.1197/jamia.M3000. Epub 2009 Aug 28. PubMed PMID: 19717802; PubMed Central PMCID: PMC3002133.

29. American Diabetes Association Consensus Panel, "Guidelines for Computer Modeling of Diabetes and its Complications (Consensus Statement)," *Diabetes Care*, 27: 2262–2265, 2004.

30. J. Hippisley-Cox, C. Coupland, et al, "Predicting risk of Type 2 Diabetes in England and Wales Prospective Deriviation and Validation of QDscore," *BMJ* 2009:380:b880.

31. B. Balkau et al., "Predicting Diabetes-Clinical, Biological, and Genetic Approaches: the D.E.S.I.R Study," *Diabetes Care*, American Diabetes Association, 31 (10), pp. 2056-2061, August 18, 2008. doi:10.2337/dco8-0368. http://www.hal.inserm.fr/inserm-00301915.

32. P. M. Clarke, A. M. Gray, A. Briggs, A. J. Farmer, P. Fenn P, R. J. Stevens, D. R. Matthews, I. M. Stratton, R. R. Holman, "A Model to Estimate the Lifetime Health Outcomes of Patients With Type 2 Diabetes: The United Kingdom Prospective Diabetes Study (UKPDS) Outcomes Model (UKPDS no. 68), *Diabetologia* 47(10), pp. 1747–1759, 2004.

33. D. M. Eddy and L. Schlessinger, "Archimedes: A Trial-Validated Model of Diabetes," *Diabetes Care*, 26(11), pp. 3093–3101, 2003.

34. D. M. Eddy and L. Schlessinger, "Validation of the Archimedes Diabetes Model," *Diabetes Care*, 26(11), pp. 3102–3110, 2003.

35. H. Zhou et al, "A Computer Simulation Model of Diabetes Progression, Quality of life, and Cost," *Diabetes Care*, 28(12), pp. 2856-2863, 2005. doi:10.2337/diacare 28.12.2856

36. M. Stern, et al, "Validation of Prediction of Diabetes by the Archimedes Model and Comparison With Other Predicting Models," *Diabetes Care*, 21(8), pp. 1670-1671, Aug. 2008.

37. R. J. Stevens, V. Kothari, A. I. Adler, I. M. Stratton, R. R. Holman, "The UKPDS Risk Engine: A Model for the Risk of Coronary Heart Disease in Type II Diabetes (UKPDS 56), *Clin Sci (Lond)*, 101:671–679, 2001.

38. R. J. Stevens, R. L. Coleman, A. I. Adler, I. M. Stratton, et al, "Risk Factors for Myocardial Infraction Case Fatality and Stroke Case Fatality in Type 2 Diabetes (UKPDS 66), *Diabetes Care*, 27:201–207, 2004.

39. A. I. Adler, R. J. Stevens, S. E. Manley, et al, "Development and Progression of Nephropathy in Type 2 Diabetes: the United Kingdom Prospective Diabetes Study (UKPDS 64), *Kidney Int*, 63:225–232, 2003.

40. I. M. Stratton, E. M. Kohner, S. J. Aldington, et al, "Risk Factors for Incidence and Progression of Retinopathy in Type II Diabetes Over 6 Years From Diagnosis (UKPDS 50), *Diabetologia*, 44:156–163, 2001.

41. A.J. Hayes, J. Leal, A. M. Gray, R. R. Holman, P. M. Clarke, "UKPDS Outcomes Model 2: A New Version of a Model to Simulate Lifetime Health Outcomes of Patients with Type 2 Diabetes Mellitus Using Data From the 30 year United Kingdom Prospective Diabetes Study (UKPDS 82), *Diabetologia*, 56(9):1925-33, Sep. 2013. doi: 10.1007/s00125-013-2940-y. Epub 2013 Jun 22. PubMed PMID: 23793713.

42. M. Vaxillaire, et al, DESIR Study Group, "The Common P446L Polymorphism in GCKR Inversely Modulates Fasting Glucose and Triglyceride Levels and Reduces Type 2 Diabetes Risk in the DESIR Prospective General French

Population." *Diabetes*, 57(8):2253-7, Aug. 2008. doi: 10.2337/db07-1807. Epub 2008 Jun 12. PubMed PMID: 18556336; PubMed Central PMCID: PMC2494697.

43. R. Jaziri, et al, DESIR Study Group, "The PPARG Pro12Ala Polymorphism is Associated With a Decreased Risk of Developing Hyperglycemia Over 6 Years and Combines With the Effect of the APM1 G-11391A Single Nucleotide Polymorphism: the Data From an Epidemiological Study on the Insulin Resistance Syndrome (DESIR) Study," *Diabetes*, 55(4):1157-62, April 2006. PubMed PMID: 16567542.

44. M. Vaxillaire, et al, "Impact of Common Type 2 Diabetes Risk Polymorphisms in the DESIR Prospective Study," *Diabetes*, Vol. 57, pp. 244-254, Jan. 2008.

45. F. A. Sonnenberg and J. R. Beck, "Markov Models in Medical Decision Making: A Practical Guide," *Med. Decis. Making*, 13:322-338, 1993.

46. J. Barhak, et al, "Chronic Disease Modeling and Simulation Software," J. Biomed. Inform. 43(5): 791-799, Oct. 2010. doi:10.1016/j.jbi.2010.06.003.

47. M. L. Puterman, *Markov Decision Processes*, Wiley, 2005.

48. O. Alagoz, et al, "Markov Decision Processes: A Tool for Sequential decision Making Under Uncertainty," tutorial paper, *Med. Decis. Making*, Jul-Aug. 2010, pp. 474 – 483.

49. V. Scmidt, *Markov Chains and Monte-Carlo Simulation*, Ulm University, Institute of Stochastics, Lecture Notes, 2010.

50. L. M. Maillart, et al, "Assessing Dynamic Breast cancer Screening Policies," *Operations Research*, Vol. 56, No. 6, pp. 1411-1427, Nov-Dec. 2008. doi 10.1287/opre.1080.0614

51. J. F. Slejko, et al, "Dynamic Medication Adherence Modeling in Primary Prevention of Cardiovascular Disease: A Markov Microsimulation Methods Application," *Value in Health*, Vol. 17, pp. 725-731, 2014.

52. K. Sigman, "Simulating Markov chains," *Notes on MC simulation*, 2007.

53. S. Yang, Y. Zhou and J. S. Baras, "Compositional Analysis of Dynamic Bayesian networks and Applications to Complex Dynamic System Decomposition," *Proceedings of the Conference on Systems Engineering Research (CSER'13)*, pp. 167-176, Atlanta, GA, March 19-22, 2013.

54. S. Yang, B. Wang, J. Baras, "Interactive Tree Decomposition Tool for Reducing System Analysis Complexity", *Proceedings of the Conference on Systems Engineering Research (CSER'13)*, pp. 138 – 147, Atlanta, GA, March 19-22, 2013.

55. American Diabetes Association http://www.diabetes.org/diabetes-basics/?loc=db-slabnav

56. CDC Centers for Disease Control and Prevention – Diabetes Home http://www.cdc.gov/diabetes/home/index.html

57. CDC 2014 National Diabetes Statistics Report http://www.cdc.gov/diabetes/pubs/statsreport14/national-diabetes-report-web.pdf

58. National Diabetes Information Clearinghouse (NDIC) http://diabetes.niddk.nih.gov/

59. Effective Health Care Program; Medicines for Type 2 Diabetes http://www.effectivehealthcare.ahrq.gov/ehc/products/155/721/OralMedT2Diab_consumer.pdf

60. D. Bertsekas, *Dynamic Programming and Optimal Control*, Vol. 1, 3rd Edition, 2005, Vol. 2, 4th Edition, 2012, Athena, MA.

61. R. A. Howard, *Dynamic Programming and Markov Processes*, Technology Press, 1960

62. S. Dreyfus, "Richard Bellman on the Birth of Dynamic Programming," *Operations Research*, Vol. 50, No. 1, pp. 48-51, Jan-Feb. 2002.

63. E. Müller, S. Maxion-Bergemann, et al, "EAGLE Diabetes Model: Basic Features and Internal Validation of Simulating Long-term Diabetic Outcomes and Related Costs," (Abstract). *Value Health*, 7:745, 2004.

64. E. Müller, et al, "EAGLE–Economic Assessment of Glycemic Control and Longterm Effects: A Computer Simulation Model for Diabetes Mellitus Type 1 and Type 2. *Diabetologia,* 47:A355, 2004.

65. E. Mueller, et al, "Development and Validation of the Economic Assessment of Glycemic Control and Long-Term Effects of diabetes (EAGLE) Model, *Diabetes Technol Ther*, 8:219–236, 2006.

66. P. Zhang, X. Zhang X, J. Brown, D. Vistisen, et al, "Global Healthcare Expenditure on Diabetes for 2010 and 2030," *Diabetes Res Clin Pract.*, 87(3):293-301, Mar. 2010. doi: 10.1016/j.diabres.2010.01.026. Epub 2010 Feb 19. Erratum in: Diabetes Res Clin Pract. 2011 May;92(2):301. PubMed PMID: 20171754.

67. M. Sordo, T. Hongsermeier, V. Kashyap, and R.A. Greenes, "A State-Based Model for Management of Type II Diabetes," *Advanced Computational Intelligence Paradigms in Healthcare (1)*, pp.27-61, 2007.

68. Roz D. Lasker, M.D. *N Engl J Med*, 1993, 329:1035-1036 September 30, 1993, and "The Diabetes Control and Complications Trial/Epidemiology of Diabetes Interventions and Complications," (DCCT/EDIC) Study Research Group, *N Engl J Med* 2005; 353:2643-2653 December 22, 2005.

69. The CDC Diabetes Cost-Effectiveness Group, "Cost-Effectiveness of Intensive Clycemic Control, Intensified Hypertension Control, and Serum Cholesterol Level Reduction for Type 2 Diabetes," JAMA, Vol. 287, pp. 2542-2551, May 2002.

70. T. J. Hoerger, et al, "Screening for Type 2 Diabetes Mellitus: A Cost-Effectiveness Analysis," *Annals of Internal Medicine*, Vol. 140, No. 9, pp. 689-710, May 2004.

71. T. J. Hoerger, "Using Costs in Cost-Effectiveness Models for Chronic Diseases – Lessons from Diabetes," *Medical Care*, Vol. 47, No. 7, Suppl. 1, pp. S21 – S27, 2009.

72. V. A. Fonseca, "Defining and Characterizing the Progression of Type 2 Diabetes," *Diabetes Care*, Vol. 32, Suppl. 2, pp. S151 – S156, Nov. 2009.

73. J. Barhak, et al, "Chronic Disease Modeling and Simulation Software," *J. Biomed Inform.*, 43(5):791-799, Oct. 2010.

74. W. Gao, “Mechanism-based Disease Progression Modeling of Type 2 Diabetes in Goto-Kakizaki Rats,” *J. Pharmacokinet. Pharmacodyn*., 38(1):143-162, Feb. 2011.

75. R. M. McCabe, “Validating a Computational Model of Patient Illness: The SIMCARE Patient Model,” *Ph.D. Dissertation*, Univ. of Minnesota, July 2012.

76. X. Wang, D. Sontag and F. Wang, “Unsupervised learning of Disease Progression Models,” *Proceedings KDD’14,* Aug. 24-27, 2014, New York.

77. D. A Propp, “Successful Introduction of an Emergency Department Electronic health Record,” *West. J. of Emerg. Med*., Vol. XIII, No. 4, pp. 358 - 361 , Sept. 2012.

78. J. King, et al, “Clinical Benefits of Electronic Health Record Use: National Findings,” *Health Services Research*, 49;1, Part II, pp. 392-404, Feb. 2014.

79. D. W. Bates, et al, “Effect of Computerized Physician Order Entry and a Team Intervention on Prevention of Serious Medication Errors,” *JAMA*, 280(15):1311-1316, Oct. 1998.

80. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 1996.

81. M. E. Cox and D. Edelman, “Tests for Screening and Diagnosis of Type 2 Diabetes,” Clinical Diabetes, Vol. 27, No. 4, pp. 132-138, 2009.

82. G. Hawe, “Scalarizing Cost-Effective Multi-objective Optimization Algorithms Made Possible with Kriging,” *COMPEL*, 27(4), pp. 836-844, 2008.

83. P.R. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification and Adaptive Control*, Prentice Hall, 1986.

84. I. M. Katsipis and J. S. Baras, “A System Engineering Framework for Multi-Criteria Performance Evaluation of Healthcare Process and Systems,” *Proceedings 5th Annual Workshop on Health IT and Economics (White 2014),* Alexandria, VA, Oct. 2014.

85. I. M. Katsipis and J. S. Baras, "Health Care Management System for Diabetes Mellitus: A Model-based Systems Engineering Framework," Summary and Poster, to appear in *Proceedings and Exhibition Cambridge Healthtech Institute's Medical Informatics World Conference*, Boston, May 2015.

86. I. M. Katsipis and J. S. Baras, "A Modeling, Health Information Technology, Analytics Framework for Healthcare Management and Application to Diabetes II", to appear in *Proceedings Healthcare Informs 2015*, Nashville, July, 2015

87. I M. Katsipis and J S Barsa "Health Care Management System for Diabetes Mellitus: A Model-based Systems Engineering Framework" , to appear in *Proceedings 26th International Conference on Software & Systems Engineering and their Applications (ICSSEA 2015),* Telecom, Paris, May 2015.

88. V. H. Poor, *An Introduction to Signal Detection and Estimation*, second Edition, Springer 1998.

89. "American Diabetes Association Standards of Medical Care in Diabetes-2012," *Diabetes Care 2012 (Supp 1),* 2012