APPLICATION OF AN INNOVATIVE MBSE (SYSML-1D) CO-SIMULATION IN HEALTHCARE

A Thesis

Submitted to the Faculty

of

Purdue University

by

Kalpak Kalvit

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Mechanical Engineering

May 2018

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Hazim El-Mounayri, Chair

Department of Mechanical Engineering

Dr. Sohel Anwar

Department of Mechanical Engineering

Dr. Ajay Thukral

Department of Mechanical Engineering

Approved by:

Dr. Sohel Anwar

Head of the Graduate Program

Dedicated to Pappa, Aai, Mayuri, Ajji and Ajoba

ACKNOWLEDGMENTS

I would like to thank my Thesis Committee members Dr. Hazim-El Mounayri, Dr. Ajay Thukral and Dr. Sohel Anwar for their support and guidance throughout the thesis work. I express my gratitude towards Dr. Vijay Thukral for his constant support and valuable suggestions in the thesis work. I would also like to thank Dr. Saulius Pavalkis for helping me understand Cameo Systems Modeler and solving my doubts and difficulties regarding the same. I would like to thank my lab mates, Hosein, Mohamed, Shashank and Amey for their support. Finally, I want to thank my family and friends (especially Prasad, Piyush, Rajat and Pental) who have been there for me through this phase.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

Figure Page

ABBREVIATIONS

MBSE	Model Based Systems Engineering
SE	Systems Engineering
OOSEM	Object Oriented Systems Engineering Methodology
BDD	Block Definition Diagram
IBD	Internal Block Diagram
STM	State Machine Diagram
ER	Emergency Room
ED	Emergency Department
GUI	Graphic User Interface
UML	Unitary Modeling Language
SysML	Systems Modeling Language
LOS	Length of Stay

ABSTRACT

Kalvit, Kalpak M.S.M.E., Purdue University, May 2018. Application of an Innovative MBSE (SysML-1D) Co-Simulation in Healthcare. Major Professor: Hazim El-Mounayri.

Contemporary Systems Engineering problems are becoming increasingly complex as they are being handled by geographically distributed design teams, constrained by the objectives of multiple stakeholders, and inundated by large quantities of design information. According to the principles of model-based systems engineering (MBSE), engineers can effectively manage increasing complexity by replacing document-centric design methods with computerized, model-based approaches. The Object Management Group (OMG) has developed a Systems Modeling Language (SysML) which is a dialect of UML 2.0. This visual modeling language provides a comprehensive set of diagrams and constructs for modeling many common aspects of systems engineering problems (e.g. system requirements, structures, functions, and behaviors). To model the behavior of complex dynamic systems in terms of co-simulation, we need to integrate SysML with advanced engineering and analysis tools.

The parametric approach in SysML captures the constraints of the block in terms of mathematical expressions. Though SysML is not intended to solve these expressions independently, they can be solved by a wide variety of solvers integrated with SysML. The core of this work is the establishment of modeling, co-simulation and analysis capabilities that do not exist in SysML independently. In this thesis, case study of Modeling of Hospital Emergency room (Door to Room Section) co-simulated with MATLAB as a solver is used for demonstration. The case study shows patient arrival, quick registration and room allocation using behavioral models in SysML. The behavioral models replicate the door to room process by co-simulating with MAT-

LAB. The proposed work shows the potential application and capabilities of SysML using 1D co-simulation in healthcare.

1. INTRODUCTION

1.1 Background

The Object Management Group has developed a graphical modeling language which is an extension of UML for application in the field of systems engineering. This modeling language basically supports construction of models in terms of their structure and behavior. It also supports co-simulation of these models up to a certain extent. SysML does not support co-simulation of dynamic systems independently. Hence it needs to be integrated with solvers that support dynamic systems. This research shows 1D Co-simulation of a case study of Door to Room of Emergency department. Emergency room is a complex system with a continuous dynamic time-behavior. In this research the author proposes a novel approach of modeling and co-simulating such systems using SysML-MATLAB Co-simulation.

The emergency department of the hospital plays an important role in providing frontline care to the severely ill and injured patients. The emergency departments in the United States are responsible for providing healthcare for over 120 million patients per year [1]. The increasing volume of Emergency Department (ED) patient arrivals and limited hospital resources is leading to increased delays and decreased patient satisfaction [2]. From 1994 to 2004, visits to hospital emergency departments increased from 93.4 million to 110.2 million-an 18 percent jump. Meanwhile, the numbers of hospitals, hospital beds, and emergency departments have declined significantly. This has led to increased burden on the resources [3]. These reasons have contributed in crowding of the patients and prolonged length of stay. Crowding and prolonged length of stay have been thought as the byproduct of inadequate number of beds, but inefficient patient care at the hospital arrival and admission may the real reason behind this scenario [1]. This inefficiency of the emergency healthcare delivery is lack of systems understanding, higher implementation costs for making changes and experimenting.

1.2 Literature Review

Systems engineering has been a promising concept that has been used for industries for decades. INCOSE defines Systems Engineering an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem. Systems Engineering concept has various life cycle models that define an approach for implementing the SE concepts (VEE diagram, Waterfall model and the Spiral model). The most famous and widely used model amongst these is the Vee Diagram developed by Forsberg at al. (2005). As shown in Figure 1.1, the VEE diagram is a generic model that can be used for design of any system. It gives a structured approach to design the system from its concept development to its disposal.

The left side of the VEE diagram starts with defining and decomposing the system while the left side is more about implementation, recomposition and Integration of the system. Systems Engineering approach is widely used in the defense and aviation industry and has proved to improve quality and value. Although other industries have accepted the advantages of using the systems approach, healthcare has been slow to adopt this methodology. Healthcare is a complex and an interdependent system. Due to this complexity and lack of understanding of healthcare system as on system, Systems Engineering approach is necessary. A systems approach to health is one that applies scientific insights to understand the elements that influence health outcomes; models the relationships between those elements; and alters design, processes, or policies based on the resultant knowledge in order to produce better health at lower cost [4]. The effective patient care depends on integration and interaction

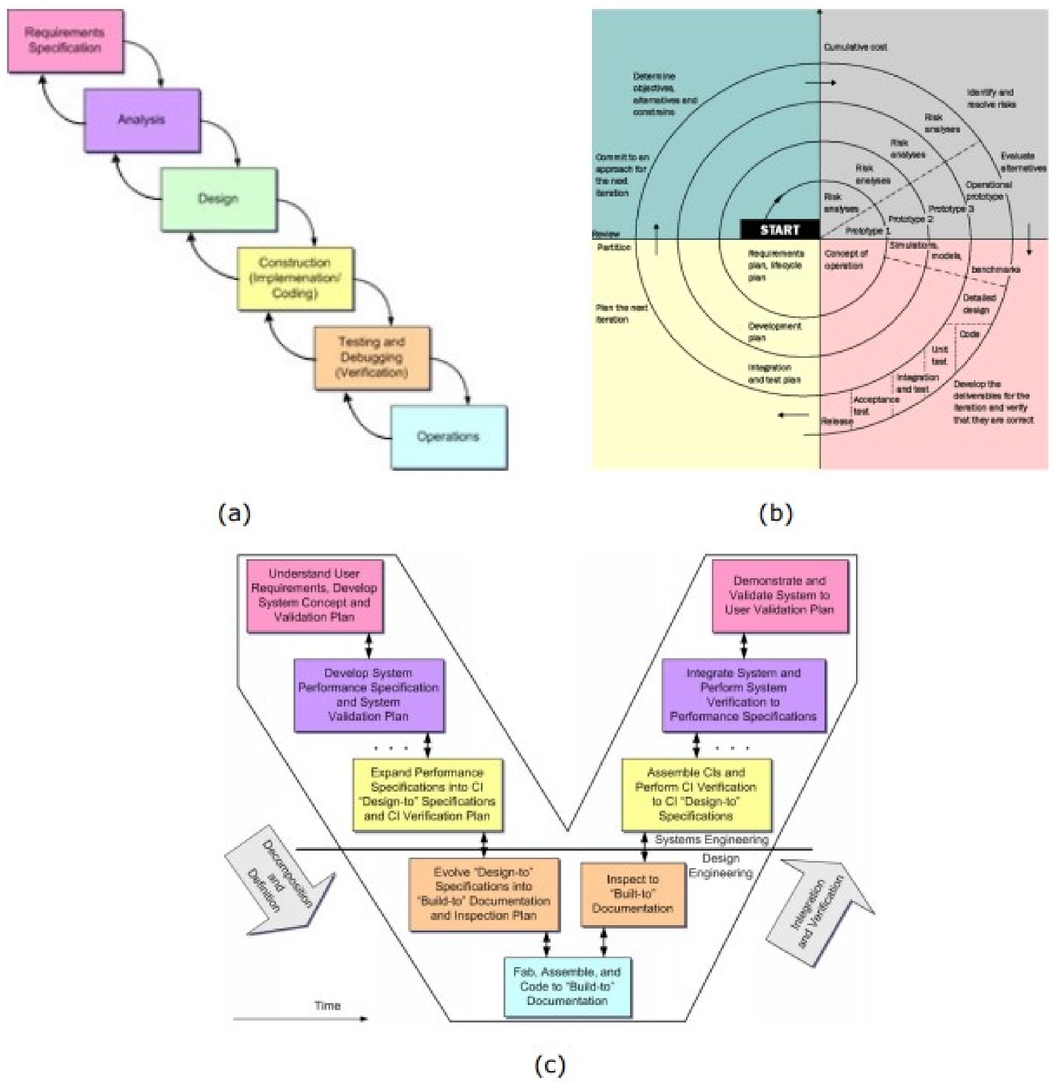

Figure 1.1. Vee model

between various systems. Even if a single entity of a sub-system is performing its function poorly, it might lead to a critical situation. In healthcare a system is a set of possibly diverse entities (patients, nurses, physicians, etc.), each performing some set of functions. The interaction of these entities as they perform their various functions gives rise to a global system behavior [5]. As specified in the VEE diagram, the traditional systems approach is about decomposing and recomposing the model. This hierarchical decomposition has worked well for other industries. But healthcare system being complex adaptive [6], dynamic properties of the system need to be considered. Healthcare system is an event-based system [5]. The behavior of this system changes with occurrence of various events. These events might be arrival of patients, change of shift physicians etc. Hence to conclude, traditional systems approach cannot be successfully applied to the systems which are non-linear and dynamic and where there is no single point of control [6]. A System such as health care system being a complex event based system will require engineering tools with SE methodology to design, analyze or manage its life cycle.

1.3 Model Based Systems Engineering (MBSE)

1.3.1 Model Based Systems Engineering Definition

Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases [7].

Increasing technological advancement in data storage, computational abilities along with emphasis on key principles of Systems Engineering have significantly increased the practice of MBSE over the traditional systems engineering approach. MBSE is a term that predicates the use of modelling to analyze and document key aspects of the Systems Engineering Lifecycle, as shown 1.2. It is broad in scope, spanning the SE lifecycle and covering levels from System of Systems to individual components. Model Based Systems Engineering (MBSE) methodology emerged from the goal of capturing the working of the system as a single, unified model rather than series of static disconnected documents [8]. MBSE is structured approach which provides methods for measuring complex systems in form of models that include software, hardware and personnel in order to improve its efficiency and effectiveness. MBSE

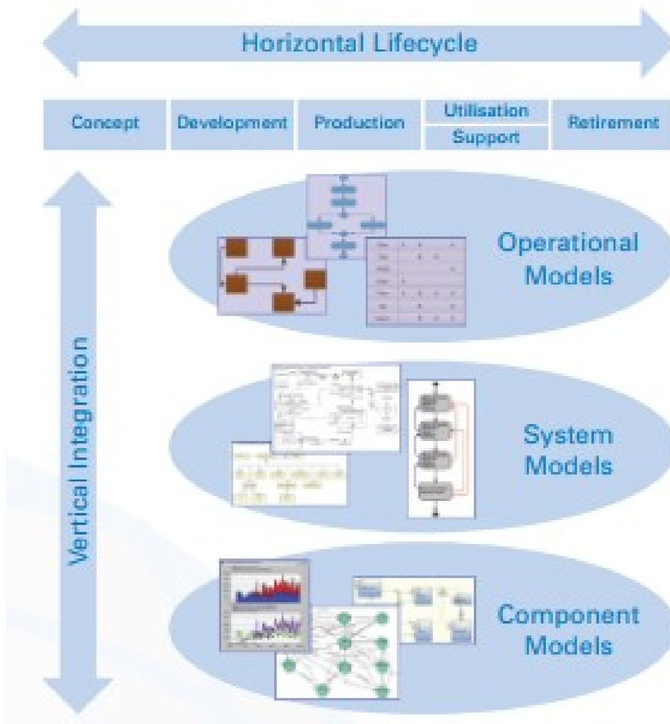

Figure 1.2. MBSE Development Life Cycle

integrated with analysis tools can prove to be very useful in terms of addressing a complex system.

The use of MBSE leads to following advantages among all the stakeholders (1) Enhanced communication due to representing the system in form of models. (2) Reduced development risk (3) Increased quality (4) Increased productivity (5) Leveraging models across the life cycle of process or product and (6) Enhanced and easy knowledge transfer.

This model-centric approach, which main artifact is a coherent model of the system being developed, contrasts with the traditional document-based one [9]. The emergence of computers in the 1950s and 1960s has strongly contributed to this paradigm shift in a considerable range of engineering disciplines like the mechanical and the electrical ones, but in the SE field the transitioning process, while becoming prevalent, is still immature [10], [9], [11]. As pointed out by Bahill and Botta [12], as a fundamental principle of good system design, the essence of MBSE relies on the application of appropriate formal models to a given domain. In the next decade, it is expected that MBSE will play an increasing role in the practice of SE and that will extend its application modeling domains beyond hardware and software systems, including social, economic, environmental, and human performance components [4], [11].

1.3.2 MBSE Methodologies

Model Based Systems Engineering is an approach and it requires a methodology i.e. collection of related processes, methods, tools and environment to apply or implement [11]. There are four leading methodologies, (1) IBM Telelogic Harmony (2) Vitech Model-Based System Engineering (MBSE) Methodology (3) IBM Rational Unified Process for Systems Engineering (RUP SE) for Model Driven Systems Development (MDSD) and (4) INCOSE Object-Oriented Systems Engineering Method (OOSEM) Unlike the most of methodologies mentioned above, OOSEM allows to captures the concepts with systems engineering approach. The core tenets of OOSEM include recognized practices essential to systems engineering that include: 1) Integrated Product Development (IPD), essential to improve communications, and 2) a recursive Vee lifecycle process model that is applied to each multiple level of the system hierarchy as shown in 1.3

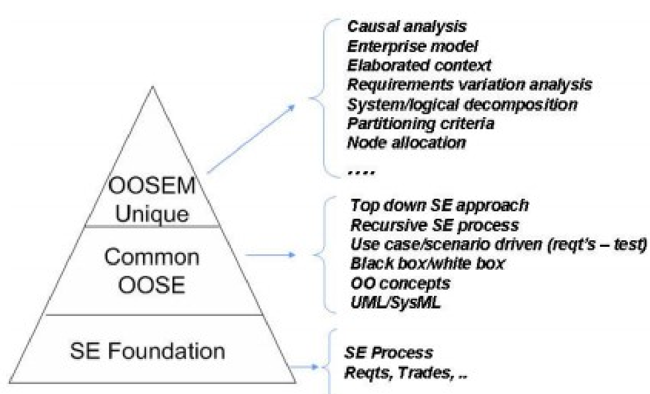

Figure 1.3. Recursive Vee Lifecycle Process Model

OOSEM includes following development activities that resemble to traditional systems engineering approach: Analyze Stakeholder Needs (2) Define System Requirements (3) Define Logical Architecture (4) Synthesize Candidate Allocated Architectures (5) Optimize and Evaluate Alternatives (6) Validate and Verify System. OOSEM commonly uses Unitary Modeling Language (UML) or Systems Modeling Language (SysML) (Developed by OMG) as modeling language. SysML is one dialect of UML. For the purpose of this project OOSEM using SysML has been chosen due to its capability to manage, process and analyze data in integration with other engineering tools which allows to utilize the full potential of the MBSE approach. OOSEM combines the traditional systems engineering approach and SysML to form a recursive System Development Process Model as shown in 1.4

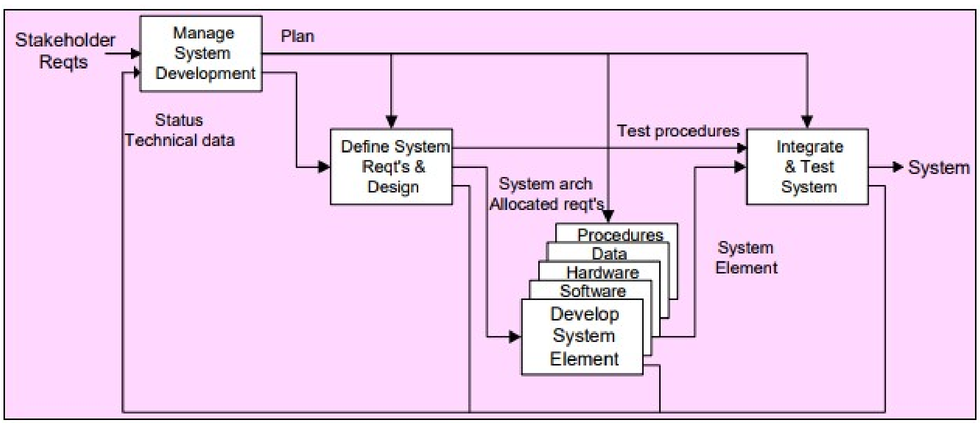

Figure 1.4. System Development Process Model

The systems engineering field has developed many tools and techniques to build and manage complex systems in defense, aerospace, and infrastructure. These are specifically intended to manage systems with many form and function elements, their interactions, and their mapping. In recent years, the Systems Modeling Language (SysML) [13], [14] has emerged as a highly popular tool embedded within the systems engineering process [15]. It is a graphical modeling language that facilitates the description of system function, form, and their allocation. It also supports parametric descriptions that may be used to develop quantitative models in coherent and wellorganized ways. Therefore, SysML is chosen as an appropriate analytical tool in this work to model the complex, interconnected, and event based nature of healthcare system.

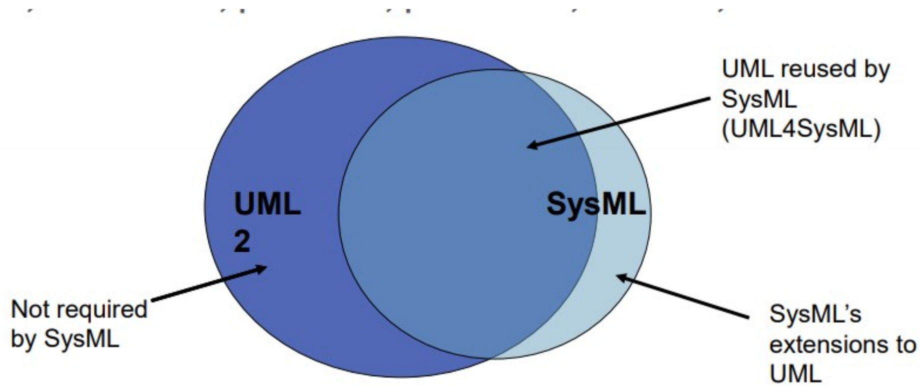

Figure 1.5. SysML as a Dialect of UML

SysML offers multiple diagrams to model the system. The diagrams are distributed over four pillars of systems engineering, namely Structure, Behavior, Parametric and requirements as shown in the 1.6. Detailed description of all the diagrams is given in the chapter of Methodology.

1.3.3 Co-Simulation in MBSE using SysML

Co-simulation models are widely used in the industry to evaluate the system and take decisions. Increasingly complex systems have developed a need of advanced engineering co-simulation tools. Whilst there are many engineering co-simulation and analysis tools, they do not provide structured approach like MBSE-SysML which is necessary for multidisciplinary and complex systems. SysML provides co-simulation and analysis capabilities only up to a certain extent. It does not have potential to execute complex mathematical expressions independently. Hence such complex models need to be co-simulated with engineering analysis tools such as MATLAB, SIMULINK, JAVA, PYTHON etc. In [16] the author shows that MBSE model integrated with MATLAB and ModelCenter provides a cohesive and consistent source

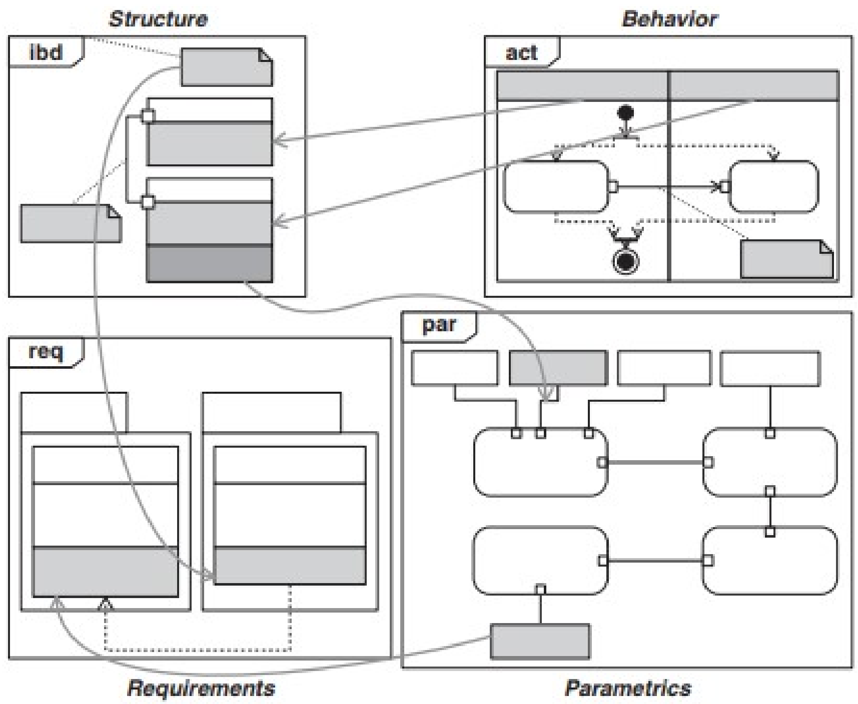

Figure 1.6. Pillars of Systems Engineering

of system requirements, design, analysis and verification. The author shows logical and physical models built using SysML and the analysis and verification using SysML integrated with MATLAB and ModelCenter. The work shows utility of MBSE tools in integration with engineering analysis tools for designing ever increasingly complex systems. In [17] the author emphasizes on improving the MBSE-SysML support by combining modeling constructs from SysML and Modelica. The paper focuses on Co-simulation of parametric diagrams in SysML and their mapping in Modelica.

To improve the support of co-simulation in SysML, many have used a MATLAB-SysML or Modelica -SysML integrated environment. The previous work in the related are shows that these integrations or co-simulations have been carried out in parametric modeling of the system [18]. In contrast to the previous work this research focuses on integrated SysML-MATLAB co-simulation in the behavior modeling of the system.

1.3.4 MBSE in Healthcare

Model Based Systems Engineering is a systems engineering concept that shifts the focus of the systems engineering effort from a document based paradigm to one in which the system information is stored, manipulated, and analyzed completely within a computer environment. By maintaining all of system information within computer models, information sharing is made seamless and the consistency of that information throughout the design process is easier to maintain. In addition, traceability can be maintained from requirements to system design, analysis, and verification. MBSE provides the systems engineer an integrated set of tools to plan and communicate the device design. In spite of many advantages of MBSE, diagramming in emergency healthcare has been constrained to flowchart based descriptions [10].

As a single diagram cannot cover all the perspectives of a complex system such as the healthcare system, there has been a need of representing healthcare system in a better manner [8]. Using MBSE, descriptive as well as computational models can be created which allows to get a brief overview of the system as well as to analyze the system by simulating it. MBSE methodology allows to capture the requirements, structure and behavior of the system in the form of models. The PCAST report 2014 focuses on use of Systems Engineering for a better healthcare delivery. It suggests using systems engineering methods such as modeling and co-simulation, statistical process control, lean techniques etc. While traditional systems engineering approach is a document based approach, Model based Systems engineering methodology gives a new dimension to systems engineering [19].

1.3.5 Limitations and Gaps in Applying MBSE to Model Healthcare System and its Delivery Process

Current applications of MBSE in healthcare delivery system lack of comprehensive and rigorous MBSE framework. The MBSE models which are typically operated by the SysML have not been used in integration with numerical tools to predict the outcomes of the process and optimize it. [16] In his research of Integrated Model-Based Systems Engineering (MBSE) Applied to the Co-simulation of a CubeSat Mission, shows the capability of MBSE using SysML integrated with analysis tools. By integrating SysMl with Matlab they have proved the fact that MBSE using SysML can be used to carry out complex analysis and run their co-simulations.

While the defense, automotive and aviation industries have used MBSE approach with SysML as a powerful co-simulation analysis tool, Healthcare industry still uses it as a diagramming tool. (Ola G. Batarseh et al, 2013) [20] in his research of SysML for conceptual modeling and co-simulation for analysis: a case example of a highly granular model of an emergency department has used SysML only for the purpose of conceptual modeling, while the co-simulation and analysis has been don using some other language and approach. Though his approach helped in getting the desired results, I believe that Modeling and Co-simulation using MBSE-SysML can prove to be more effective and the possibility of risk errors can be minimum.

Dirk Zwemer, Intercax LLC in his research has developed a time based drug delivery model using SysML. In his research he has used third party plugins such as Syndeia and ParaMagic to connect SysML to Database management tools and analysis software to show that a model linking the system architecture, databases of patient and facility data, and a powerful math solver allows us to explore alternative treatment scenarios and improve healthcare resource utilization [8]. Though this research demonstrated the immense potential of MBSE using SysML, I believe that the need for using third party plugins was not necessary

1.3.6 Thesis Objective

Simulation and Co-Simulation for analysis and verification of the system have been used in SysML in the parametric modeling. Though carrying out simulation in the parametric modeling satisfies the purpose of analysis and verification, it does not satisfy the purpose of using MBSE methodology. The parametric modeling executes actions such as data transfer, evaluation of mathematical expressions, updating the block values and properties. However, it does not show the order in which these actions have been executed. The objective of this thesis is to show the optimum point of simulation or co-simulation as the behavioral modeling in contrast to the conventional parametric simulations. The purpose behind simulating the system in behavior modeling is that it gives a better understanding of the system in terms of order of execution of actions and flow of data.

MBSE and SysML in healthcare delivery has been used as a more of diagramming tool and not for co-simulation and analysis. The objective of this thesis is to describe a methodology and a framework that utilizes recent advances in MBSE and associated tools, to develop a conceptual architecture as well as to prove that MBSE approach applied in integration with the right engineering tools can prove to be a powerful co-simulation and analysis tool. It was not the goal of thesis to develop a system that can be brought into immediate use, rather it is not possible without accurate data and guidance from the hospital faculty. The goal is to demonstrate potential of MBSE approach used with modeling language SysML with integration of Analysis and database management tools in emergency healthcare delivery. Emergency healthcare is an integration of multiple complex systems and the first or the initial part of an emergency department is registration and room allotment. Hence, effective management of resources in the registration and room allotment section is critical as it can be leading factor in contributing for increased length of stay and crowding of the patients.

This thesis is focused on the arrival section of the emergency department, also classified as Door to Room section. It covers patient arrival, registration and dispatch of patient in the emergency room. The objective was to create a structural and behavioral model of the Door to Room section of a generic emergency department, simulate this model in real time to see the various states and the corresponding time each resource spends on a particular patient and the time required by the patient to get to the emergency room after arrival at the hospital. The part of data (Arrival data) for this project was provided by the largest healthcare facility in the state i.e. Sidney and Louis Eskenazi Hospital. Some part of data was obtained from Center for Disease Control (CDC). While not all the required data could be obtained, some data had to be generated based on assumptions. Assumptions and data generation has been explained in the methodology chapter. Due to lack of data in some cases and due to some synthesized data, the thesis does not have accurate results in terms of numerical value. But surely, it does show that if this approach and the developed model are linked to real life data, its full potential and value can be appreciated. The developed model shows the flow trends and resource utilization, which will help the hospital administrators to streamline the process to use the resources efficiently and effectively to decrease the possible wait time of the patients

1.3.7 Thesis Outline

This thesis describes an innovative framework for MBSE (SysML) 1D Co-simulation. It describes the tradition approach for co-simulating SysML model and then focuses on the new approach and its advantages over the traditional approach. It uses a case study of emergency department to show the proposed framework.

In chapter 2 the methodology behind the MBSE approach is explained. It also describes about the methodology used for generation and synthesis of test data. The MBSE framework and architecture using the MagicGrid approach has been described in this chapter. It also speaks about SysML as a modeling language. In chapter 3 the implementation of the methodology described in chapter 2 has been described in detail. It shows the generation of data and various diagrams of created using SysML. It also speaks about the workflow of the process and Cameo Co-simulations Modeler as a modeling and co-simulation tool.

Chapter 4 focuses on the results and discussions. It shows the verification processes used and speaks about the case study used to simulate the Door to Room behavior. Chapter 5 is conclusion. It summarizes the work and focuses on the objectives achieved from this work. It also speaks about the limitations of the current work and the future work needed in order to overcome these limitations.

2. METHODOLOGY

This chapter describes about the methodology and approach used for creating a co-simulation model for the Healthcare Case Study. The initial part describes about the MBSE theory and approach used for this research, this part is divided in two sections (1) MBSE architecture and SysML as the modeling language and (2) MBSE framework using Magic Grid approach. The second part describes about the methodology used for generating test data and data synthesis which was necessary for the defining physical characteristics of the model and executing the co-simulation successfully.

2.1 MBSE Theory

Model-based SE (MBSE) is an emerging approach in the SE field [21], [22], and can be described as the formalized application of modeling principles, methods, languages, and tools to the entire lifecycle of large, complex, interdisciplinary, sociotechnical systems. A simplified definition of MBSE is provided by Mellor et al [23] as ...is simply the notion that we can construct a model of a system that we can transform into the real thing. [11]. There are many programming languages such as Simula67, Object Modeling Technique (OMT) and Unitary Modeling Language (UML). These languages lack the characteristics of systems of SE, support aspects and trade studies [11]. Hence in a joint effort by INCOSE and OMG an new dialect of UML was developed only for SE : Systems Modeling Language (SysML). SysML provides graphical representation of the data in model repository. It supports design, analysis, specification and verification of complex systems. It is considered as de facto of modeling language for SE [24]. The approach to apply MBSE using SysML for Door to Room of the Emergency department was developed from a generic approach for modeling systems by No Magic. This approach is called as the Magic Grid approach.

It is based on four pillars of SE i.e. Structure, requirement, parametric and behavior and the corresponding nine diagrams using SysML. The modeling and co-simulation of the system was created by Cameo Co-simulations Modeler by No Magic integrated with MATLAB and Microsoft excel. This chapter shows in detail the methodology applied for modeling and co-simulation of Door to Room behavior in the Emergency department.

2.2 MBSE Architecture

As mentioned above Magic Grid framework has been used to develop the model from requirements to co-simulation. The four pillars for Magic Grid approach are shown in 2.1 [9]

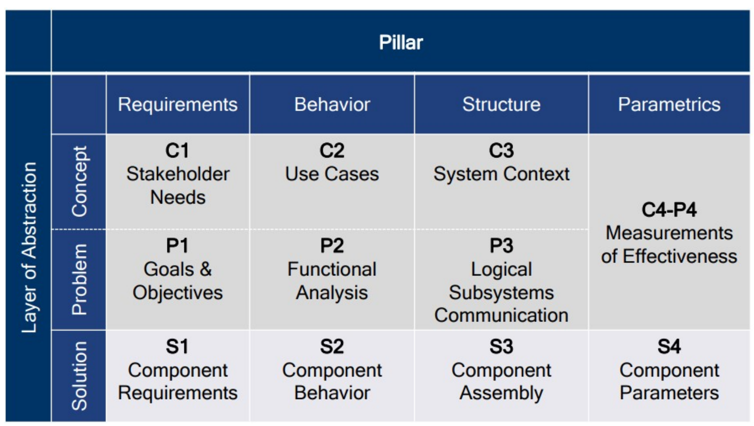

Figure 2.1. Model Framework

SysML diagrams for development of Door to Room behavior were developed on the basis of the four pillars specified in the figure. There are total 9 SysML diagrams and all of them were implemented in this research. The nine SysML diagrams are shown in 2.2

All the nine diagrams mentioned in 2.2 are defined as follows:

Requirement diagram: represents text-based requirements and their relationship with other requirements, design elements, and test cases to support requirements

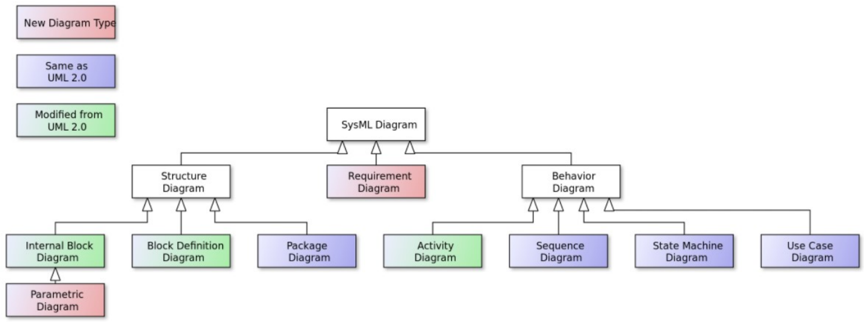

Figure 2.2. Types of SysML Diagrams

traceability (not in UML). In the context of this thesis, requirements diagram or requirements table defines the requirements or the functions that the developed tool shall be capable of executing

Activity diagram: represents behavior in terms of the order in which actions execute based on the availability of their inputs, outputs, and control, and how the actions transform the inputs to outputs (modification of UML activity diagram). Every state in the behavior of Door to Room has an activity diagram that represents actions executed each activity of the state, it also has a activity diagram of the whole system just for representational purposes. It does not have any function in particular.

Sequence diagram : represents behavior in terms of a sequence of messages exchanged between systems, or between parts of systems (same as UML sequence diagram). Sequence diagram is usually generated automatically depending upon the defined order of behavior of the system. It shows the messages exchanged between the system internally amongst its entities. In this case, the sequence diagram was generated after completely defining the behavior of the system.

State machine diagram : represents behavior of an entity in terms of its transitions between states triggered by events (same as UML state machine diagram). In context to this thesis, the state machine diagram represents states of each resource throughout their working shift/hours. Each state machine diagram has a do/entry/exit behavior.

Each of these behaviors is an activity diagram which depicts the actions taking place in that particular state.

Use case diagram : represents functionality in terms of how a system is used by external entities (i.e., actors) to accomplish a set of goals (same as UML use case diagram). There are a total of 3 Use case diagrams, each diagram shows some part of the ED as a system which is used by the patient as an actor of the system. These three use cases are then combined into one single use case that shows the interaction of the actor with various entities of the system.

Block definition diagram: represents structural elements called blocks, and their composition and classification (modification of UML class diagram). Block definition diagram for the door to room system shows the elements of the system and their nature of relationship with the system, it also defines the behavior and the properties that each block possesses.

Internal block diagram : represents interconnection and interfaces between the parts of a block (modification of UML composite structure diagram). Internal block diagram in the thesis shows the patient flow from on element of the arrival section to other. Due to some limitations of SysML on the internal block diagrams, composite structure diagrams were developed for co-simulation purposes. These limitations and justification for using composite structure diagrams is explained in the results section.

Parametric diagram : represents constraints on property values, such as F [m*a], used to support engineering analysis (not in UML). Parametric diagrams are used to evaluate mathematical expressions. In this thesis most of the mathematical expressions have been evaluated using activity diagrams using opaque behaviors and opaque actions. So the behavioral part of the system is doing the parametric. This part is explained in detail in the co-simulation and algorithm section.

2.3 Magic Grid Approach/Framework

The MBSE approach for this project is derived from the Magic Grid approach by No Magic. The flow of this approach is as follows: This section describes the Magic Grid framework in context of the thesis: Requirements Pillar: Stakeholders Needs: Stakeholder needs basically captures the requirements, regulations and polices defined by the stakeholder. The Stakeholder in Door to room context is going to be a system analyst or a hospital system administrator to provide with the requirements. These requirements are refined and then put in a form of diagram or a table using SysML.

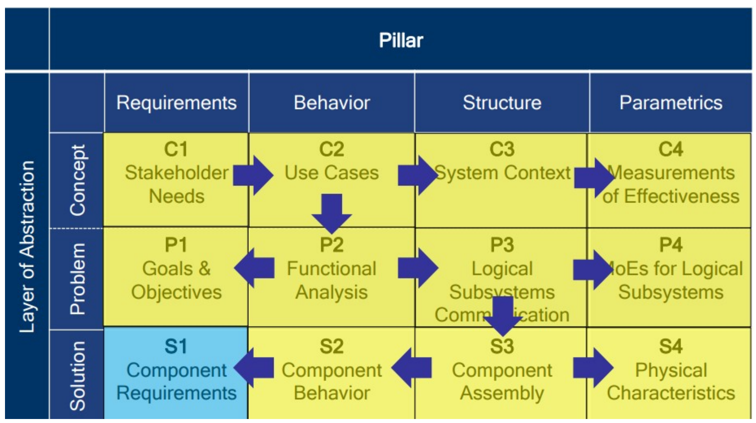

Figure 2.3. Flow of Magic Grid Framework

A) Requirements Pillar:

Goals and guidelines: It explains the planned achievement. The goals are generalized and long term such as in this case creation of SysML tool for analysis of Door to Room behavior. Objectives are more specific and planned. They are the tasks needed to complete in order to achieve the goal.

Component requirements: components come in the lower level of the system hierarchical structure. Component requirements are the subsystem requirements needed to achieve the system goal. They are also derived from system requirements. In the context of this research Arrival section (Door to room) is considered as the system and registration, room allotment and their subsequent actors can be considered as sub systems. The requirements derived from the arrival section for registration or any other sub system is the component requirement.

B) Behavior Pillar:

Use cases: Use case is used to capture the user needs or the actor as defined in SysML in the form of use case specification as well as additional information such as the flow of information or object. As mentioned earlier there are three use cases, particularly giving the overview of patient interaction and flow from Door to Room.

Functional analysis: Functional analysis can be defined as the behavior of the subsystems. It shows the internal object and information flow in the subsystems. These are generally represented by activity or state machine diagram.

Component behavior: It defines the detailed behavior of the components of sub-system. It has the algorithm/ call operation actions and signals. In this research, Quick registration is a functional analysis of the Registration process and transferring the data to Emergency Medical records (EMR) is the component behavior.

C) Structure Pillar:

System Context: It shows the high level interfaces needed by the system to communicate with the systems in the environment. In this case the user of the system or the actor such as the patient requires the Registered nurse to interact with the Emergency Department (System).

Logical Subsystems Communication: Logical architecture represents an intermediate abstraction between functional and physical architecture. Components of a logical architecture represent abstractions of physical solutions.

Component Assembly: Shows physical connections based on physical interfaces between physical systems/subsystems/components. Generaly shown by Block Definition Diagrams and Internal Block Diagrams. As this research is process oriented, Component assembly in this context is the connection between different elements like registration, room allotment, patient etc.

D) Parametric Pillar:

The parametric pillar consist of measures of effectiveness and the physical characteristics of the system. The system is generally simulated with the help of parametric diagrams to get the results from the mathematical relationship between the physical characteristics of the system. This part of the Magic Grid approach has been combined with the behavioral part. The physical properties are defined in the structure of the system and the mathematical evaluation is done in the behavioral pillar using opaque behaviors and opaque actions. The opaque behavior or opaque action uses the physical properties of its classifier to evaluate the functions or mathematical expressions defined in its body.

2.4 Generation of Data and its Synthesis

As specified in the objective of this thesis, the aim was to simulate the model in real time and make it look as realistic as possible. In order to do so it was an important part of thesis to generate patient arrivals, profiles and resource schedules. As only some data was available online and some part obtained from Eskenazi Health, rest had to be generated and synthesized with the given data. This section gives a brief description about the methodology used for the same.

2.4.1 Methodology for Generating Arrival Data

Eskenazi hospital provided the part of arrival data for this research. This data was for the year of 2014 and 2015. For each year the data was given in the form of weekdays as rows and the columns were time of the day divided into the time interval of ten minutes. 2.4 shows the snapshot of data provided by Eskenazi in Microsoft Excel.

2014	All Patients Arrived by 1/6 Hr per Day of the week							
Row Labels	**0:00**	**0:10**	**0:20**	**0:30**	**0:40**	**0:50**	**1:00**	**1:10**	**1:20**
Sun	32	79	87	84	65	84	56	61	63
Mon	35	56	56	59	64	57	49	54	61
Tue	29	74	73	60	66	56	62	53	45
Wed	27	73	81	61	74	49	64	55	59
Thu	34	79	71	57	55	60	55	56	48
Fri	31	74	64	67	59	41	43	54	54
Sat	32	78	72	89	103	79	64	58	58
Grand Total	220	513	504	477	486	426	393	391	388

Figure 2.4. Snapshot of the Data Given by Eskenazi Health

As shown in the figure the data is divided into days of the week over the year. It shows the number of patients arrived in a ten minute interval on a day of the week throughout the year. It does not show timestamp of each patient. Timestamps of each patient are necessary to run the co-simulation in real time. In the book Applied Statistics, Principles and Examples by D.R.Cox [25], in the example of admissions in intensive care unit the author has stated that arrival of each patient is a completely random series and the events occur independently. But the cumulative number of patient arrival has a pattern, author further states that arrival rates are likely to have a specific variation based on day of the week and time of the day. These parameters are already specified in the given data. The author shows Poisson distribution for the number of patients arriving after a particular time length. This is shown in the figure below.

If we plot the Eskenazi data for any particular day, we get the same Poisson distribution. As shown in figure 2.6, Eskenazi plot follows the same distribution as described by the author in **??**. The author states that the frequency of patient arrival between these time intervals is completely random. Hence these timestamps were created using the methodology of random number generation. Random numbers can be classified in two types truly random and pseudo random numbers. Pseudo random numbers are uniformly distributed random numbers generated using deterministic formulas unlike truly random numbers that depend on physical or natural

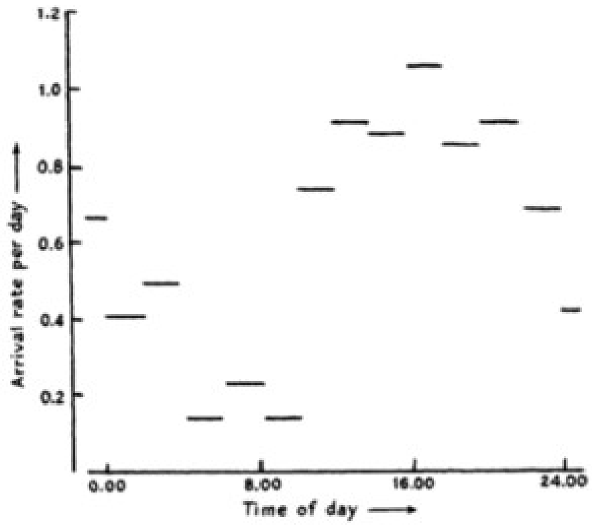

Figure 2.5. Poisson Distribution for the Number of Patients Arriving After a Particular Time Length

phenomenon like atomic decay [26]. Hence, pseudo random generation was used to create timestamps for each patient arrived in the specified interval on any given day. Therefore, we can say that each timestamp represents one patient.

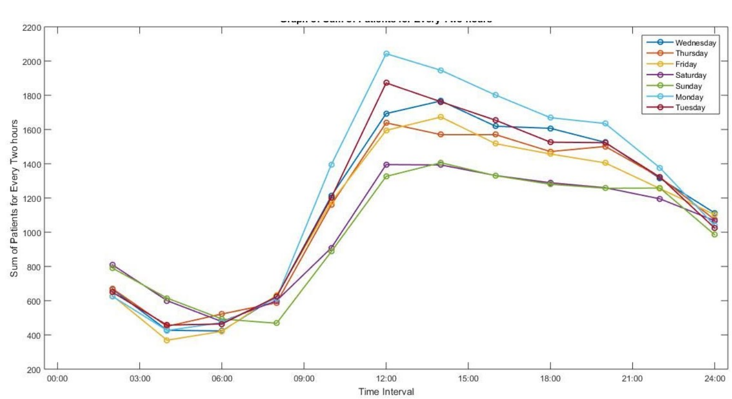

Figure 2.6. Plot of Eskenazi Data

2.4.2 Methodology for Generating Demographics Data

Patient profile was generated for the purpose of co-simulation and future scope. Patient demographics, chief complaint and acuity distribution of the patient arrival in the Emergency department was obtained from the government website Center for Disease Control (CDC). The source data is taken from National Hospital Ambulatory Medical Care Survey (NHAMCS). The data was collected from 23rd of December 2013 to 23rd of December 2014 from 319 Emergency service areas across the nation. The data is divided in three parts based on age groups and each part is divided in sections according to the sex and chief complaint. 2.7 shows the snapshot of the table, ten leading principal reasons for emergency department visits, by patient age and sex: United States, 2014. The table shows percentage distribution for each age group based on gender and chief complaint.

Principal reason for visit and RVC code[1]		Number of visits in thousands (standard error in thousands)		Percent distribution (standard error of percent)	
All visits	...	141,420	(11,464)	100.0	...
Stomach and abdominal pain, cramps and spasms	S545	11,135	(882)	7.9	(0.3)
Chest pain and related symptoms (not referable to body systems)	S050	6,887	(593)	4.9	(0.2)
Cough	S440	5,751	(733)	4.1	(0.3)
Fever	S010	5,536	(660)	3.9	(0.4)
Headache, pain in head	S210	4,296	(496)	3.0	(0.2)
Back symptoms	S905	4,120	(586)	2.9	(0.3)
Shortness of breath	S415	3,405	(328)	2.4	(0.2)
Pain, site not referable to a specific body system	S055	3,254	(317)	2.3	(0.1)
Vomiting	S530	2,936	(408)	2.1	(0.2)
Injury, other and unspecified type-head, neck, and face	J505	2,616	(342)	1.8	(0.2)
All other reasons[2]	...	91,484	(7,362)	64.7	(0.5)
All visits under age 15	...	27,739	(2,769)	100.0	...
Female	...	13,167	(1,396)	47.5	(1.0)
Fever	S010	2,165	(300)	7.8	(0.6)
Cough	S440	1,286	(214)	4.6	(0.7)
Stomach and abdominal pain, cramps and spasms	S545	742	(149)	2.7	(0.5)
Skin rash	S860	687	(113)	2.5	(0.3)
Vomiting	S530	581	(108)	2.1	(0.3)
Earache, or ear infection	S355	546	(106)	2.0	(0.3)
Injury, other and unspecified type-head, neck, and face	J505	*405	(124)	1.5	(0.4)
Symptoms referable to throat	S455	333	(74)	1.2	(0.2)
Headache, pain in head	S210	292	(77)	1.1	(0.2)
Laceration or cut of facial area	J210	*	...	*	...
All other reasons[2]	...	5,866	(664)	21.1	(1.2)
Male	...	14,572	(1,427)	52.5	(1.0)
Fever	S010	2,140	(314)	7.7	(0.7)
Cough	S440	1,514	(279)	5.5	(0.8)
Vomiting	S530	795	(171)	2.9	(0.5)
Injury, other and unspecified type-head, neck, and face	J505	597	(120)	2.2	(0.4)
Skin rash	S860	581	(93)	2.1	(0.3)
Stomach and abdominal pain, cramps and spasms	S545	468	(60)	1.7	(0.2)
Earache, or ear infection	S355	451	(108)	1.6	(0.3)
Symptoms referable to throat	S455	241	(51)	0.9	(0.2)
Labored or difficult breathing (dyspnea)	S420	230	(53)	0.8	(0.2)
Laceration or cut of facial area	J210	217	(48)	0.8	(0.2)
All other reasons[2]	...	7,339	(675)	26.5	(1.0)

Figure 2.7. Patient Profile Table

To create patient profile for each patient (timestamp) generated in the previous step, age, gender and chief complaint needs to be assigned to each patient/ timestamp. Using the percentage distributions as shown in the table, a probability tree diagram was used to find the probability of each patient profile.

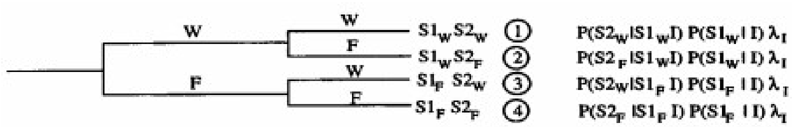

Figure 2.8. Probability Tree Diagram

The probability trees, also addressed as decision tree was used to find the probability of occurrence of event [27], [28], in this context it was used to find the probability that an event will occur where a patient with certain profile arrives at the emergency department. Such a probability tree was built for generating the patient profile. These probability distributions from the tree were then used in Matlab to get the patient profile dataset.

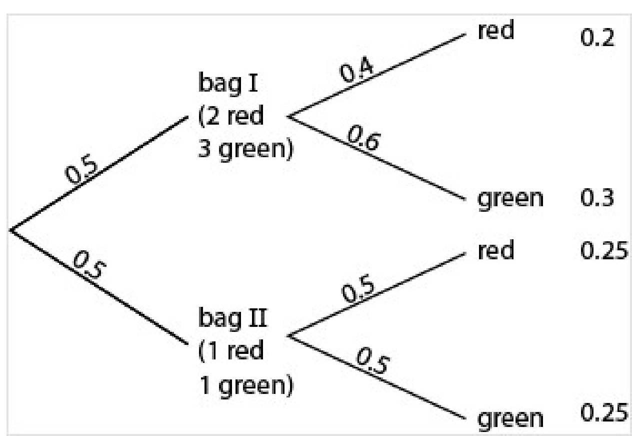

Figure 2.9. Simple Application of Probability Tree diagrams

2.4.3 Emergency Severity Index Distribution

Emergency Severity Index or the acuity is a nomenclature used to define the level of illness or the level of emergency of the patient. On registration in the emergency department, each patient is assigned with an acuity level or ESI. A generic distribution of ESI of the patients arriving in the hospital is shown in 2.10. 1 to 5 shows the level of emergency where, 1 being the highest and 5 being the lowest. The patients are transferred to Triage levels/ rooms based on their ESI. For the purpose of this problem, ESI has been defined according to the nomenclature used in the Eskenazi Emergency Department. Where, 1 = Shock Room, 2 = High Acuity 3 = Low Acuity and 4 = Intake.

1. Intake Area: Intake serves patients with minor injuries who come to the ED and leave immediately after their treatment. Low Acuity Area (LA): LA serves patients with low acuity illness who are required to stay inside the ED for a long time until their condition becomes stable. Those patients usually occupy the room on an average of 4-5 hours and sometimes for more than one day. 3. High Acuity Area (HA): HA serves patients with high acuity illness who need to stay inside the ED for a long time on an average of six hours. Shock Rooms: Patients who need immediate rescue are treated inside Shock rooms; and after getting stabilized, they are generally admitted in the hospital.

In [29] the author mentions shows that likelihood of patient having a particular ESI depends upon the Gender, Age and the Arrival time. However, in book [30] the author shows that the ESI trend is a random distribution. Hence, for the simplicity of the problem the ESI/acuity distribution has been considered as a random distribution.

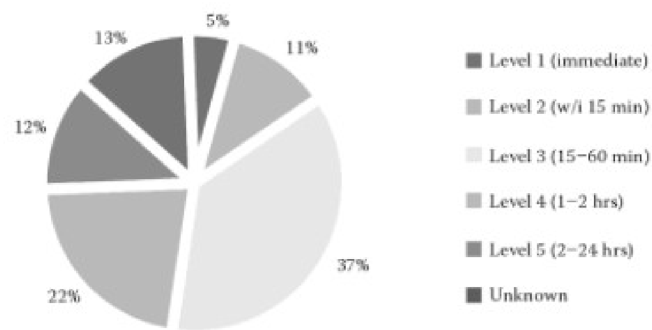

Figure 2.10. Emergency Severity Index Distribution

3. IMPLEMENTATION IN HEALTHCARE CASE STUDY (DOOR TO ROOM)

This chapter shows the implementation of the methodology described in chapter 2. The first part of this chapter focuses on the generation of test data and its synthesis. The next part discusses about Cameo Systems Modeler as a Co-simulation and Modeling tool and the last part focuses on the modeling and architecture of the Door to Room model using the SysML diagrams.

3.1 Emergency Department (Door to Room) Process Review

For this problem a simple generic ED process has been taken into consideration. The total Door to Room process has been divided in three parts. The initial part is where a virtual patient is generated based on the given data from Eskenazi and cdc.gov. This patient is assigned with a profile and arrival time. The next two parts of the process are Quick registration and Room Allotment. Quick registration is the initial registration process where patients Age, Gender, Date of Birth and Chief Complaint/ Medical reason for visit is collected and stored in the EMR. The patient is then sent to care Tech for room allotment. The care Tech checks patients Emergency Severity Index and assigns him a room if available. If the room is not available then patient is sent to the waiting area. The details of patient sent to waiting area are captured in database. When the room is available, the patient is called upon by the Care Tech by the order of waiting time and the Severity Index of the patient. Based on the ESI the patient is sent to LA, HA or Shock. This process is shown by an activity diagram in Cameo. (Note: The activity diagram shown below is just for representational purpose and does not have any behavior. Detailed activity diagrams are later shown in this chapter). The model is discussed further in this chapter with regards to the 3 parts as shown in figure 3.1

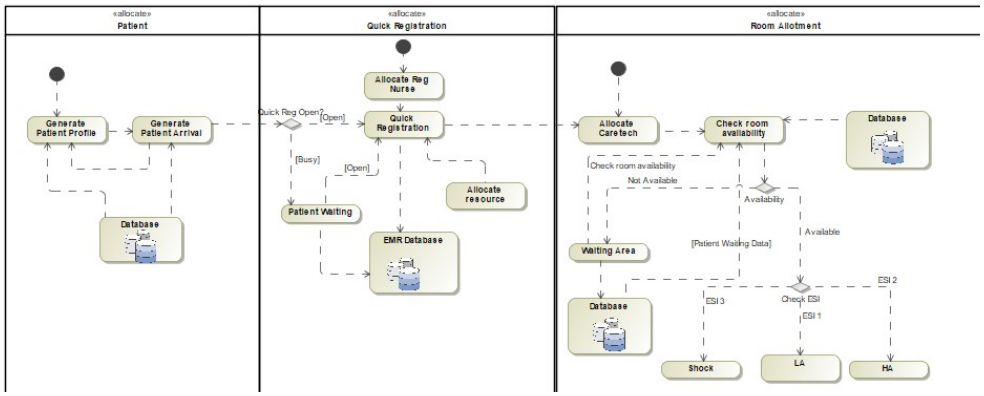

Figure 3.1. Overview of Door to Room Procedure

3.2 Generation and Synthesis of Data

This section describes about the implementation of the methodology used for generating patient arrival data.

3.2.1 Data Generation

The software used to create these timestamps in the specified interval was MATLAB. Sidney an Louis Eskenazi health has provided with arrival data for patients. They have recorded the cumulative number of patients arrived at the emergency department for the year of 2014. The data has been collected for the total number of each day of the week per year (eg. Data for Monday is cumulative data for all the 53 Mondays in the year 2014).The data consists of 7 rows which implies each day of the week and 144 columns which implies 10 minute interval of each weekday, as shown in 3.2. Methodology for creating target data: 1. The source data is imported from an excel file to Matlab. 2. As the data is for year of 2014, the time frame for creating timestamps was specified as 1st of January 2014 to 31st December 2014 3. Each day of the week/year was divided into 10 minute interval. And each interval was divided into corresponding seconds. 4. Timestamps were generated in Matlab for this time window using Random Number Generator. 5. Each cell from the source data specifies the number of timestamps to be generated for that specific time interval of day of the week. 6. These timestamps were then randomly assigned to the dates throughout the year corresponding to the weekday on that date and the time interval. 7. The final dataset (Target data) consisted of a unique timestamp for every patient arrived at Eskenazi hospital in the year 2014. 3.2 shows timestamps generated in string format in a screenshot of MATLAB command window.

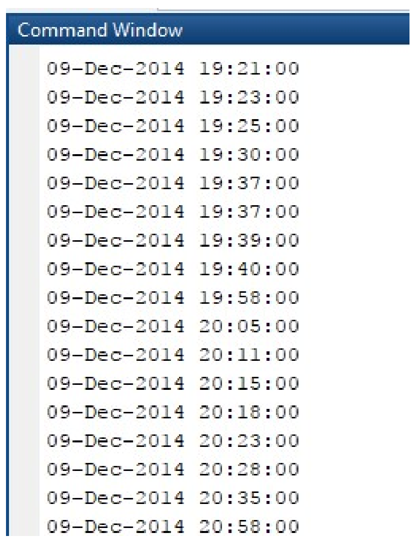

Figure 3.2. Timestamps Generated in Matlab

As shown previous section, using similar approach probability tree was built to generate the patient profile.

A snapshot of MATLAB command window for patient profile is shown in 3.4. It shows the dataset or an array created for Females under age of 15 years for chief complaint Vomiting. Such datasets were created for each case and then randomly shuffled amongst themselves using MATLAB. The datasets were created using the probability distribution found out by the tree and then were shuffled amongst themselves by using random permutations and combinations (randperm) in MATLAB

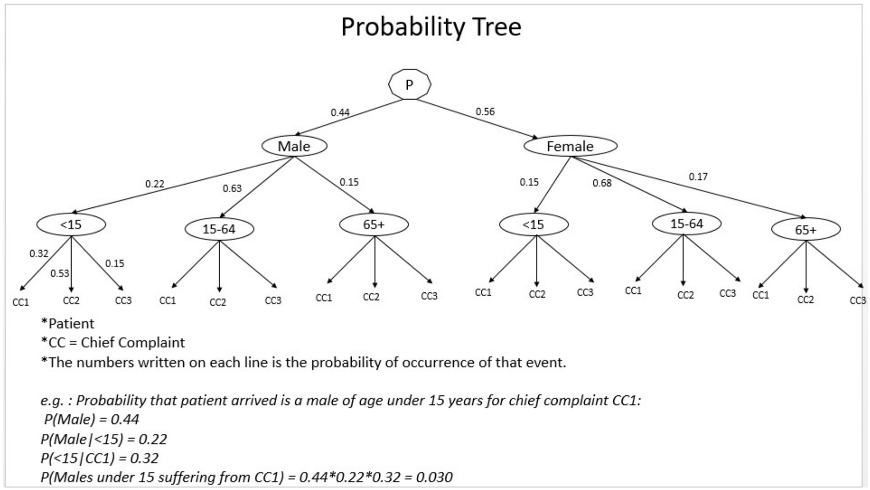

Figure 3.3. Probability Tree for Patient Profile

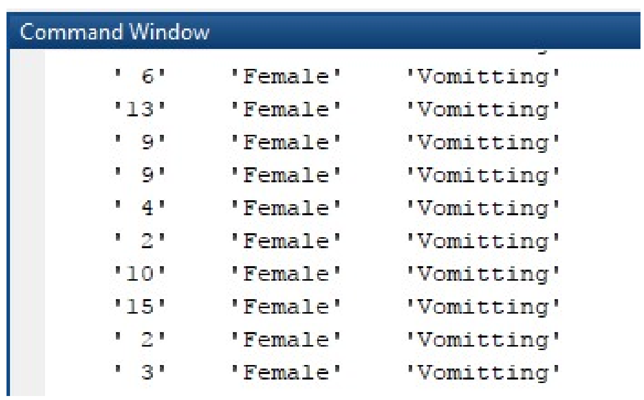

Figure 3.4. Patient Profile Generation in MATLAB

The ESI distribution was generated in MATLAB. A distribution of ESI with the percentages as shown in previous section was generated in MATLAB using Random Number Generator. These numbers were then sampled using random permutations and combination function in MATLAB to get a uniformly distributed pseudo random distribution of ESI.

3.2.2 Data Integration

Data integration is combining data obtained from several disparate sources to provide a unified view. As arrival data was from Eskenazi and the demographics were obtained from the government sources (cdc.gov), there was a need to integrate this data. As stated earlier, the demographics data was collected in a National Survey. Table 1.0 from [19] shows that almost 30 percent of the data was collected from the Mid-West region. On observing the national and the Midwestern census and comparing it with the local (Indiana) census, a lot of similarities were found. Hence the possibilities that the demographic trend for emergency arrival at the national level will be same as that of the local is high. Using this basis, the data created in different steps as mentioned above was integrated in a single table to get a complete patient profile for each patient arrived in the emergency department. 3.5 shows the part of table generated in MATLAB and then later exported in excel. This patient profile was created for all the patients arrived at the Eskenazi Emergency department for year 2014. The table shows Arrival Time in String format, Patient Age, Gender (1=Male, 2=Female), Date of Birth (DOB) and Emergency Severity Index (ESI).

3.3 Framework for the Case Study

As discussed earlier, the goal of the thesis is to show the co-simulation in the behavioral modeling itself. To achieve this, the parameters and corresponding expressions need to be evaluated in the behavioral model itself. The 3.6 shows the framework followed for achieving this objective. In the MagicGrid approach which evaluates Measures of Effectiveness (MOEs) and component parameters in the parametric modeling, the proposed framework contrasts it by evaluating the MOEs and component parameters in the behavioral pillar itself, thereby eliminating the need of parametric pillar from the MagicGrid Approach.

Arrival Time	Age	Gender	DOB	ESI
1/1/14 12:07 AM	55	1	6/6/59	3
1/1/14 12:21 AM	54	1	1/5/60	3
1/1/14 12:28 AM	87	1	1/16/27	3
1/1/14 12:29 AM	37	1	1/22/77	2
1/1/14 12:58 AM	53	0	1/1/61	2
1/1/14 1:01 AM	51	1	1/30/63	4
1/1/14 1:08 AM	65	1	1/12/49	2
1/1/14 1:15 AM	78	1	1/4/36	4
1/1/14 1:49 AM	75	1	1/3/39	2
1/1/14 2:07 AM	41	1	1/23/73	2
1/1/14 2:51 AM	89	1	1/4/25	3
1/1/14 3:04 AM	35	1	1/27/79	3
1/1/14 3:57 AM	26	1	1/26/88	1
1/1/14 4:06 AM	71	1	1/7/43	3
1/1/14 4:51 AM	55	1	1/14/59	2
1/1/14 4:59 AM	77	1	1/4/37	3
1/1/14 5:04 AM	56	1	12/31/57	3
1/1/14 5:13 AM	76	0	1/11/38	2
1/1/14 5:21 AM	87	1	1/12/27	3

Figure 3.5. Patient Profile Distribution

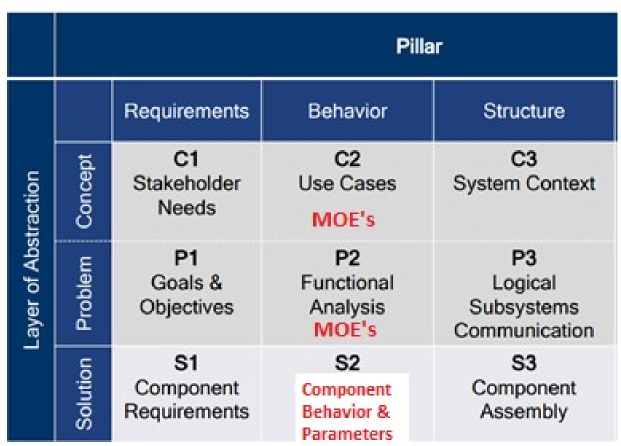

Figure 3.6. Framework for the Case Study

3.4 Cameo Systems Modeler as a SysML Tool

Cameo Systems Modeler is an industry leading cross-platform collaborative Model-Based Systems Engineering (MBSE) environment, which provides smart, robust, and intuitive tools to define, track, and visualize all aspects of systems in the most standard-compliant SysML models and diagrams. It provides Systems Engineers with continuously checking of model consistency, requirements verification and tracking progress with design metrics. Cameo Systems Modeler uses SysML which is developed on the basis of UML version 2. In this research all the framework/ architecture and for ED (Door to Room) has been developed in Cameo Systems Modeler . This tool gives a capability to integrate with many other engineering tools, in our case MATLAB and MS Excel.

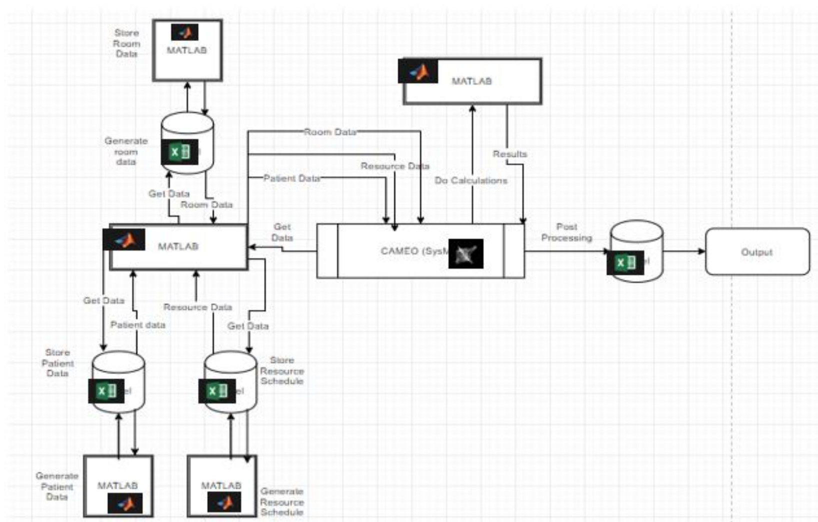

Figure 3.7. Workflow Diagram for Door to Room

3.7 shows the tools that are integrated with Cameo Systems Modeler for the purpose of this research. Cameo allows the integration of Matlab to compute mathematical parameters in order to execute the parametric of the model created. In our case the data required for co-simulation is stored in MS Excel and is imported using MATLAB. Besides this MATLAB is also used to compute equations and distributions in some cases. The following 3.8 shows Cameo Systems Modeler user interface. The containment tree is where the created elements are stored in for of packages in a hierarchical manner. Diagram pallet has all the elements that are supported for that particular diagram type. And the Diagram pane is where the diagram is actually drawn.

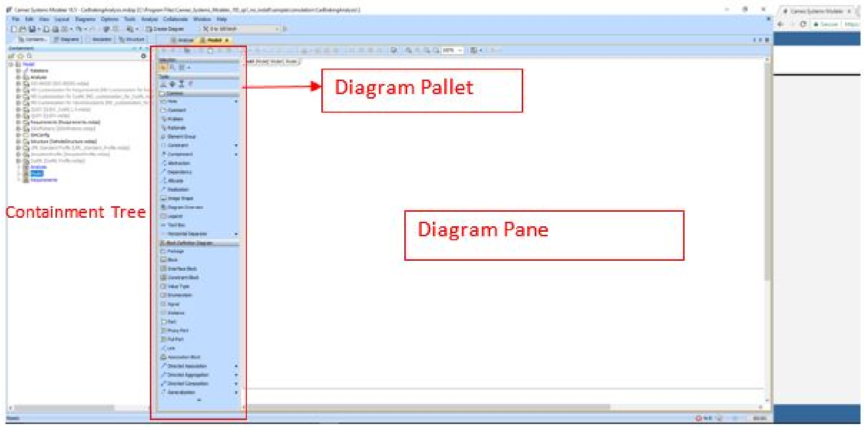

Figure 3.8. Cameo Systems Modeler User Interface

Cameo Systems Modeler allows to create three main types of project. 1. SysML Project 2. Co-simulation Project and 3. Scrum Project. SysML Project: It uses only Systems Modeling Language and is designated for Systems Engineering applications. 2. Co-simulation Project: with this type, predefined UML profiles can be used. It gives user a capability to do UI prototyping and Co-simulation and Execution configurations. 3. Scrum Project: It is a type of project with predefined artifacts and brief guidelines which allows to create user stories, requirements modeling and sprint plans. In this research SysML Project and simulation Projects have been used. SysML project has been used to create the structure, behavior and requirements of Door to Room. Co-simulation project has predefined UML profiles, moreover it supports JAVASCRIPT APIs. Execution Configuration feature in this type of project gives the capability to record results with timestamp and export to .csv* files. This feature is important in the context of this research as there are a lot of values generated in real-time for every iteration of the co-simulation.

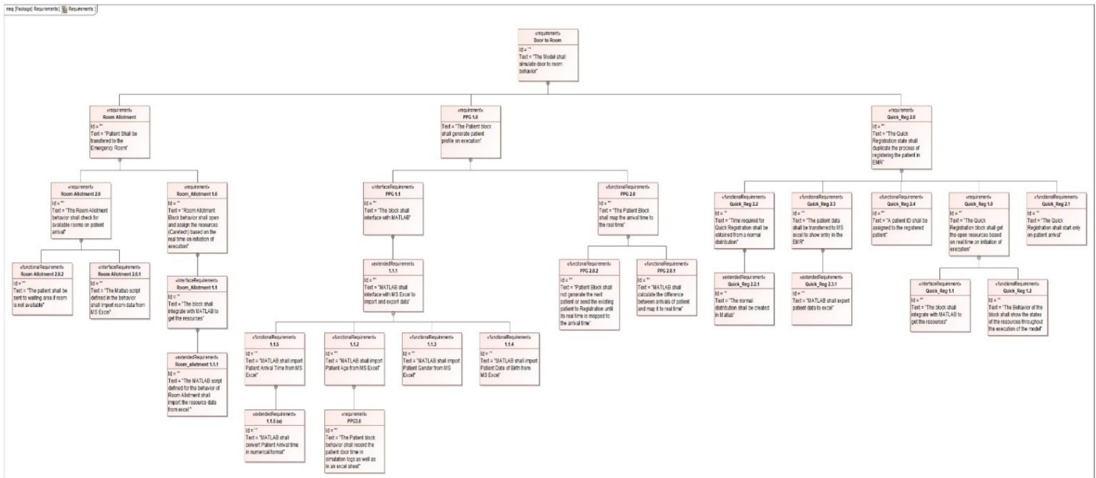

Figure 3.9. Requirements Diagram

3.5 MBSE Model Requirements

The above figure is the requirements diagram for the Door to Room Model. It has all the stakeholder needs, target goals and objectives as well as the component level requirements that need to be satisfied in order to allow the model to achieve its intended purpose. The stakeholders of this model can be considered as the Hospital Emergency Department administrators or the system analyst, in this research the author himself can be considered as analyst or the stakeholder. SysML requirement diagram gives capability to capture text based requirements and define the relationship between them. There are total eight types or subclasses of requirements out of which three are used in this diagram. These requirement types enable the modeler to define the relationship between multiple requirements as well as between the model elements. The sub-classes used in this context were 1. Functional requirements 2. Interface Requirements and 3. Extended Requirements. Functional requirements are the ones that specifies a behavior that a system or a part must perform. Interface requirements is the requirement that specifies ports for connecting system with other systems. Extended requirements are sub types of standard requirements, these add extra properties to the requirement element.

In 3.9 hierarchical decomposition of the requirements is shown from different specification is shown. The requirements are either defined by ¡¡derive¿¿ relationship or containment relation. Derived relationship corresponds to the next level in the hierarchy. It is generally used to give detailed requirements derived from the main requirements of the system. Containment relationship in the system hierarchy is generally the decomposition of the main requirement of the system. It is used to breakdown the main requirement into sub-requirements.

#	Name	Text	Derived	Verify Method	Verified By
1	1.0 Door to Room	The Model shall simulate door to room behavior		Inspection	Door to Room
2	PPG 1.0	The Patient block shall generate patient profile on execution	Door to Room	Inspection	Check Patient Profile Gene
3	PPG 1.1	The block shall interface with MATLAB	PPG 1.0		
4	1.1.1	MATLAB shall interface with MS Excel to import and export data		Inspection	Generate PA
11	PPG 2.0	The Patient Block shall map the arrival time to the real time	PPG 1.0	Inspection	-door_time : String
12	PPG 2.0.1	MATLAB shall calculate the difference between arrivals of patient and map it to real time			
13	PPG 2.0.2	Patient Block shall not generate the next patient or send the existing patient to Registration until its real time is mapped to the			
14	Quick_Reg 2.0	The Quick Registration state shall duplicate the process of registering the patient in EMR	Quick_Reg 2.1 Door to Room	Inspection	Registration(classifier beha

Figure 3.10. Requirements Table

The 3.10 shows the partial requirements table generated for this project. The model (Door to Room) has a main requirement as 'The Model shall simulate door to room behavior'. Three main requirements were derived from this requirement. These three requirements are then broken down to component level requirements. The three main requirements are as follows: The Patient block shall generate patient profile on execution, Quickreg block shall duplicate the process of patient registration on execution, Patient shall be transferred to the emergency room once in the room allotment section. Many component level requirements were derived from these requirements. The requirements table also shows the verification methods and the verifying components for most of the requirements. The verification of requirements or the verification of model is discussed in the later part of this chapter. The full requirement table is shown in the appendix. The requirements generation has helped in defining specific task, milestones, time line and the deliverable.

3.6 MBSE Model Structure

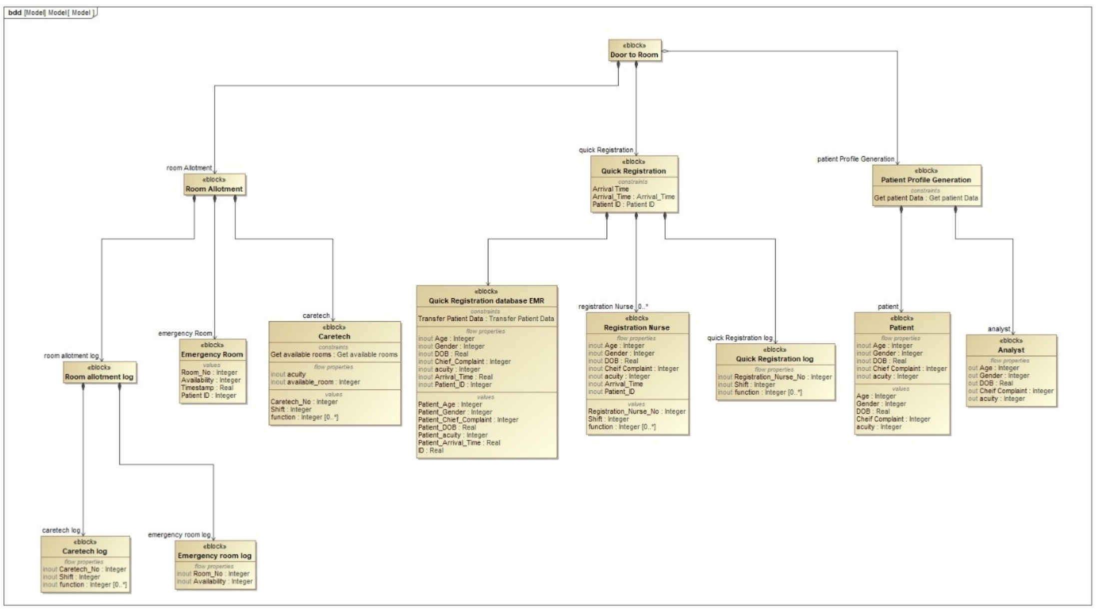

Figure 3.11. Block Definition Diagram

The figure shows Block Definition Diagram of ED process (Door to Room). The process structure is hierarchically divided in three main blocks, Emergency Room, Quick Registration and Patient Profile Generation. The relationship between the system and Emergency Room, Quick Registration blocks is composition. This shows that Emergency Room and Quick Registration are an eternal part of the system (Emergency Department). These blocks are further divided into their sub-blocks, Quick Registration is composed of three sub-blocks namely, Registration Database (EMR), Registration staff log and Registration Nurse. Each of these blocks is defined with flow properties. Flow property is a property of block to which stereotype ¡¡flow property¿¿ has been applied. It shows the flow of items or information from the block or to the block depending upon the specified direction. Block Patient Profile

Generation is connected with the door to room by aggregation relationship, it shows that Patient and Emergency department can exist without each other. It also shows that it can have one or more patients. The Door to Room Internal Block Definition Diagram is shown in 3.12. The IBD shows the flow of patient from one block to other.

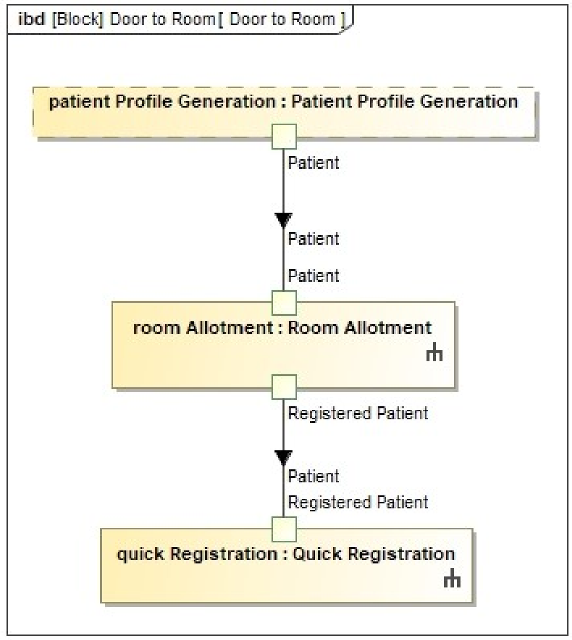

Figure 3.12. Internal Block Diagram of the System (Door to Room)

Each of the blocks and their internal structure is explained in the next sections. The Block Definition Diagram is explained by 3 main blocks as follows:

3.6.1 Patient Profile Generation

The Block Patient Profile Generation has two sub-blocks 1. Analyst and 2. Patient. It has on constraint property called Get Patient Data. The Internal Block Diagram shows that the analyst gets patient data from the constraint property. Constraint property is defined with a constraint, this constraint is a Matlab function name, this function imports the patient data from an excel file.

This data is then transferred to the analyst flow properties from the constraint parameters. The analyst block has flow ports for transferring the data to the Patient

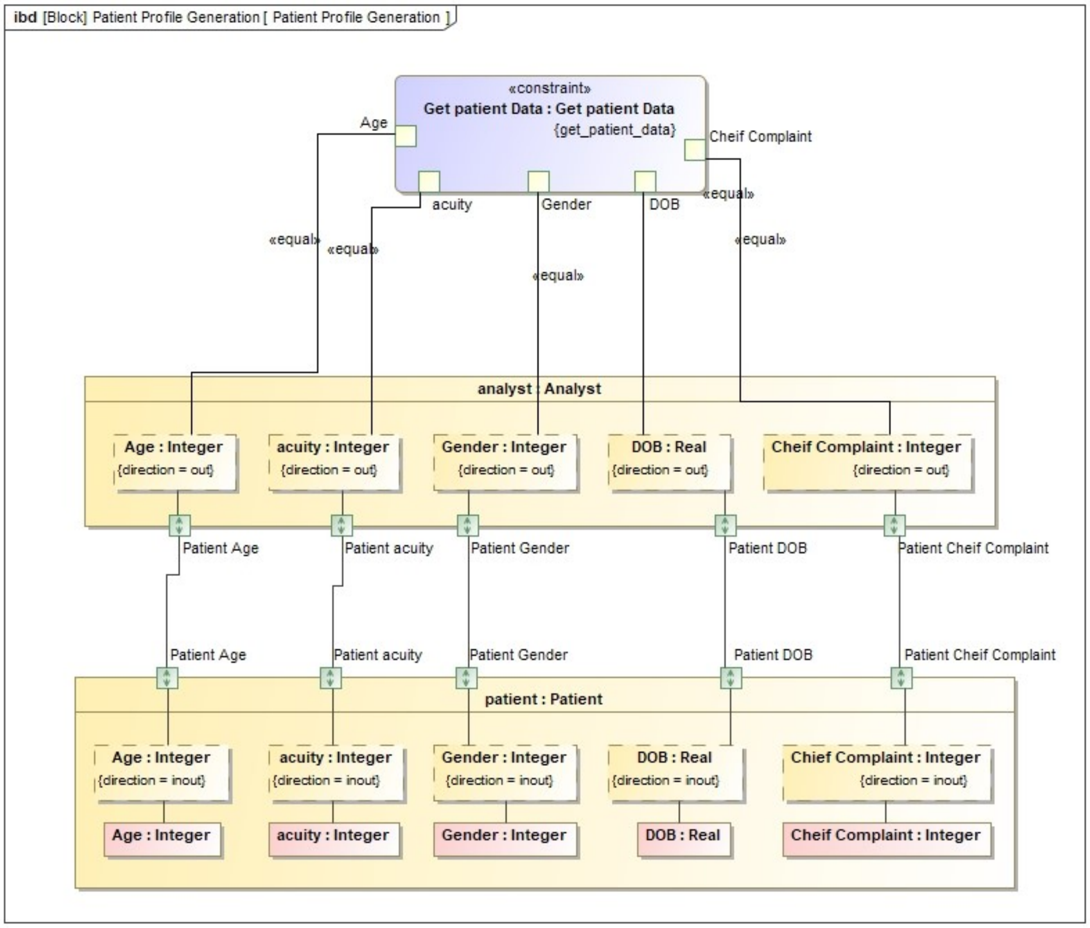

Figure 3.13. Internal Block Diagram of Patient Profile Generation

block. The data is transferred through the flow ports to the flow properties of Patient block. The data is then finally stored as value properties of patient. The data is stored as value properties as Age, Gender, acuity etc. are the attributes of the patient and need to flow with the patient as a structure. The classifier behavior of the block patient profile generation is explained in the co-simulation section.

3.6.2 Quick Registration

Quick registration is the process of registering the patient data in the Emergency Medical records. The above figure shows the information flow between the Registered

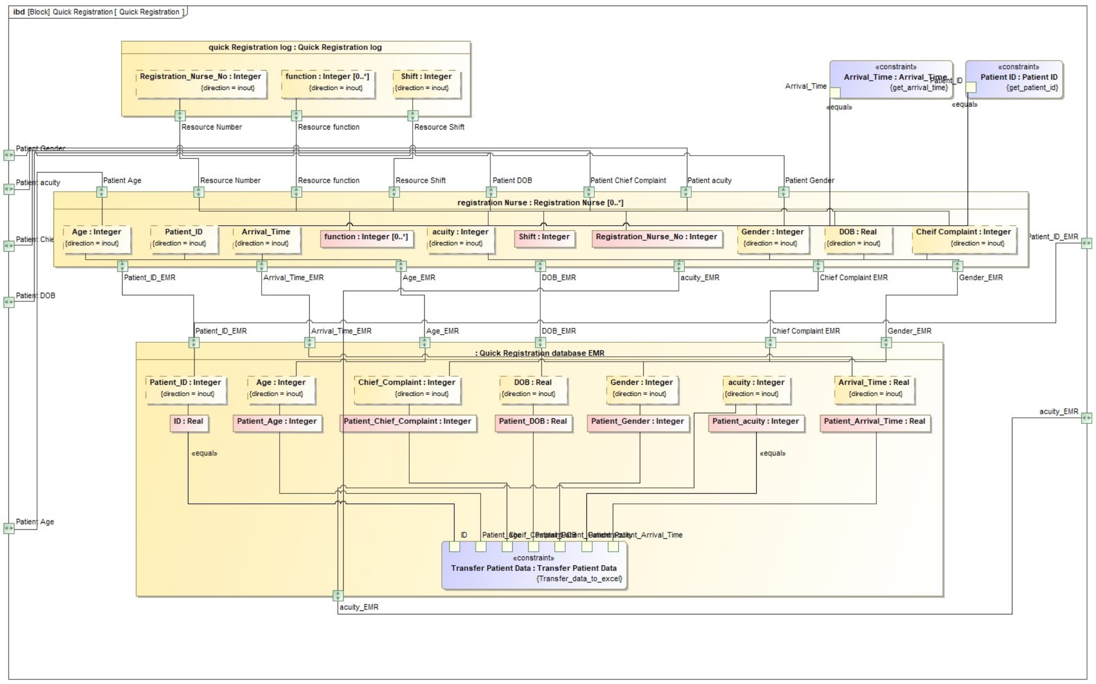

Figure 3.14. Internal Block Diagram for Quick Registration

Nurse, Patient and the Quick registration database. As shown in 3.14, the block patient flows to Quick registration from Patient Profile Generation. Once the patient arrives at quick registration its information is collected and stored in the EMR by a registered nurse, then the patient is sent to room allotment. The block quick registration has 3 main elements, the quick registration log, quick registration database and registered nurse. Each of these blocks have flow properties. Quick registration log has been assigned with flow properties Registered Nurse no, Shift and, Function id. Quick registration log has a list of resources with these properties.

These properties flow through the corresponding ports to the block Registered nurse and are stored as value properties. The ports on the diagram frame show flow of information from the patient (block: Patient Profile Generation). This information flows through the ports to the block registered nurse as flow properties. This information is then transferred by the registered nurse to EMR as value properties. The registered nurse records the patient arrival time and signs it an ID which is shown by a constraint property. These constraint properties have constraints whose language is MATLAB. 'Arrival time' calls the MATLAB function to record the current time. 'Patient ID' calls a MATLAB function to assign a 4 digit random number as a patient Id. The Block Quick Registration Databse (EMR) transfers the received patient data to an excel file. The flow property for patient acuity and patient ID flows to the room allotment block through the flow ports. The constraint property Transfer Patient Data uses MATLAB function to transfer the data to excel file. These behaviors are explained in detail in the co-simulation section.

3.6.3 Room Allotment

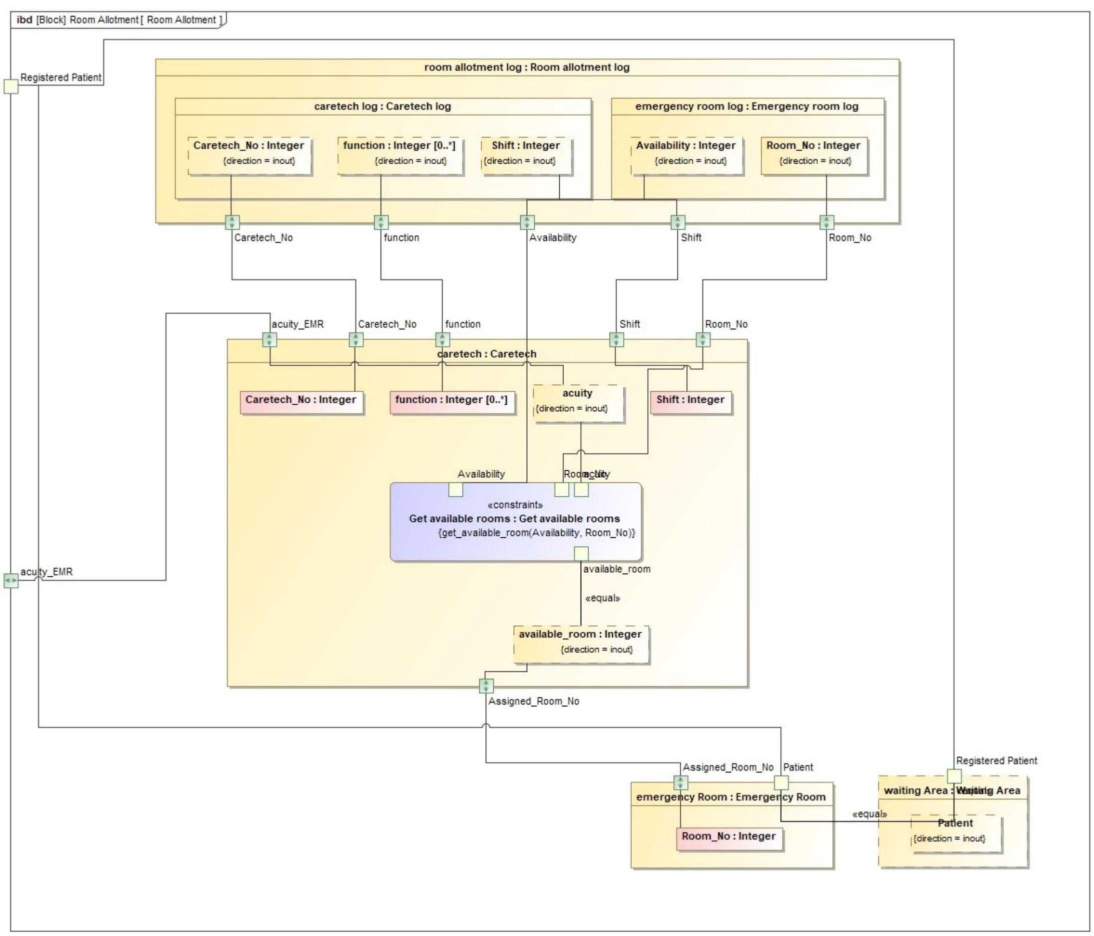

Figure 3.15. Internal Block Diagram for Room Allotment

Room allotment is the procedure of assigning the room to patient and transferring him to the room or to the waiting area. The ports on the diagram frame shows the flow of patient and the acuity of patient as recorded in the EMR to the block room allotment. The block room allotment log has two sub-blocks or elements caretech log and room allotment log. Caretech log has been defined with flow properties Caretech No. , function id, Shift which flow to the Caretech and stored as value properties. The Caretech has a constraint properties which calls a MATLAB function to get the available rooms at current time from the excel files. It selects the room based on the acuity of patient. The available room number is updated as a value property in the emergency room block. Depending upon the availability, the patient either flows to emergency room or the waiting area.

3.7 MBSE Behavior Model

3.7.1 Use Cases

Uses cases were created for all the three main sub-blocks described in the section structure. Uses cases typically define the list of actions or events that an actor undergoes or carries out in order to achieve the goal of the system. Use case diagrams for blocks Patient Profile Generation, Quick Registration and Room allotment were created. Each of the use cases are assigned with a behavior or the list of events in order to achieve the goal of that particular block. In this research, each use case was assigned with a behavior in form of state machine diagram, and further each state in the state machine diagrams have a separate behavior in the form of activity diagrams. This section gives an overview of the use cases, however the detailed behavior is explained in the subsequent section (Refer section co-simulation) The use case diagrams are as follows:

A) Case Diagram for Patient Profile Generation

This use case diagram shows the generation and allocation of the patient profile. Include relationship is used to factor out the common functionality between multiple use cases and is always performed as part of the base use case. The Use Case diagram shows four use cases, 'Generate Patient Profile' is the main use case which includes use cases 'Get Acuity', 'Get Patient arrival time' and 'Get Patient Demographics'. This use case diagram shows the generation and allocation of the patient profile.

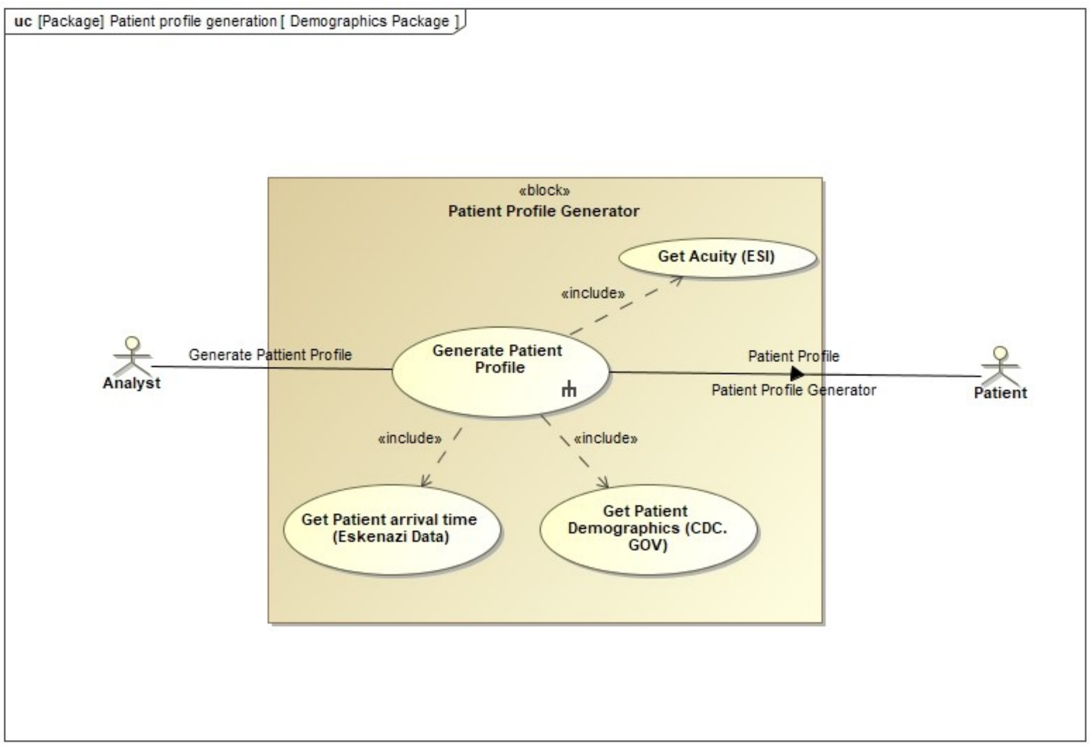

Figure 3.16. Use Case Diagram for Patient Profile Generation

Include relationship is used to factor out the common functionality between multiple use cases and is always performed as part of the base use case. The Use Case diagram shows four use cases, 'Generate Patient Profile' is the main use case which includes use cases 'Get Acuity', 'Get Patient arrival time' and 'Get Patient Demographics'. Include relationship between these use cases specifies that 'Generate Patient Profile' includes the behavior of the other two use cases. In the above figure the two actors are Analyst and Patient. Both Patient and Analyst are defined as primary actors in this case. A primary actor is the primary stakeholder of the system that acts on the system to deliver one/all of its services. In this context, Analyst initiates the generation of patient profile and allocates it to the Patient. The block Patient Profile Generator is the system boundary for the uses cases. Both Patient and the analyst are not the part of the system, association relationship defined between the actors and use cases defines the communication that occurs between the two in order to accomplish the functionality associated with the use cases.

B) Use Case Diagram for Quick Registration

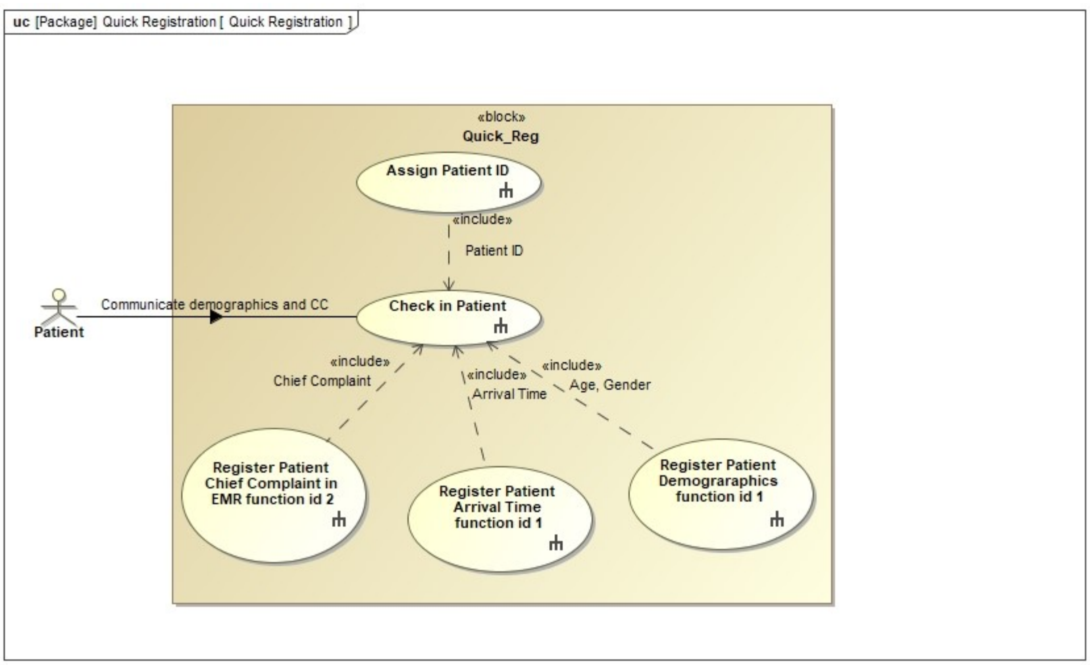

Figure 3.17. Use Case Diagram-Quick Registration

The above figure shows the use cases for Quick Registration, the base use case is Check-in Patient (Or can be called as Quick Register Patient). This use case includes multiple use cases. Each of these use cases have defined behaviors. Check in patient is the use case for registering patient which includes 'Register Patient Demographics' (For Registering the Patient data in Emergency Medical Records), 'Register Patient Arrival Time' (For Registering Patient Arrival Time in the Emergency Medical Records (EMR)), 'Register Patient Chief Complaint' (For examining Patient medical condition and register the corresponding chief complaint in the EMR) and 'Assign Patient ID' (Assigning an ID to the corresponding patient). The actor for this use case is the Patient which has an association with the base use case to show the communication of its demographics and chief complaint with the registration staff. The registration staff is an eternal part of the system and not an actor, meaning that it does not act on the system but acts within it. Hence it falls in the system boundary (In this context Quick Reg).

C) Use Case Diagram for Room Allotment

The figure below, shows the Use Cases for Room Allotment. The base use case for this context is 'Transfer patient to room'. It includes two use cases 'Assign Room', 'Check room availability' and 'Check patient's acuity'. Once the patient is done with quick registration, it goes to the room allotment section, where the acuity of the patient is checked. Acuity is assigned in the Quick Registration process, hence use case 'Check patient's acuity' has an extended relationship with the 'Check-in Patient'. Once the acuity of the patient is checked, then corresponding room availability is checked. For example If the patient is assigned with High Acuity, then the High Acuity area is checked for available rooms, this is shown by 'Check room availability'. If a room is available then it is assigned to the corresponding patient and the room database is updated, these functions are shown by the use case 'Assign room'. If a room is not available for corresponding acuity level of the patient then the patient is transferred to the waiting area, this is shown by the use case 'Transfer patient to waiting area'. This use case has an extended relationship as this functionality is optional. Every patient is not necessarily transferred to the waiting area.

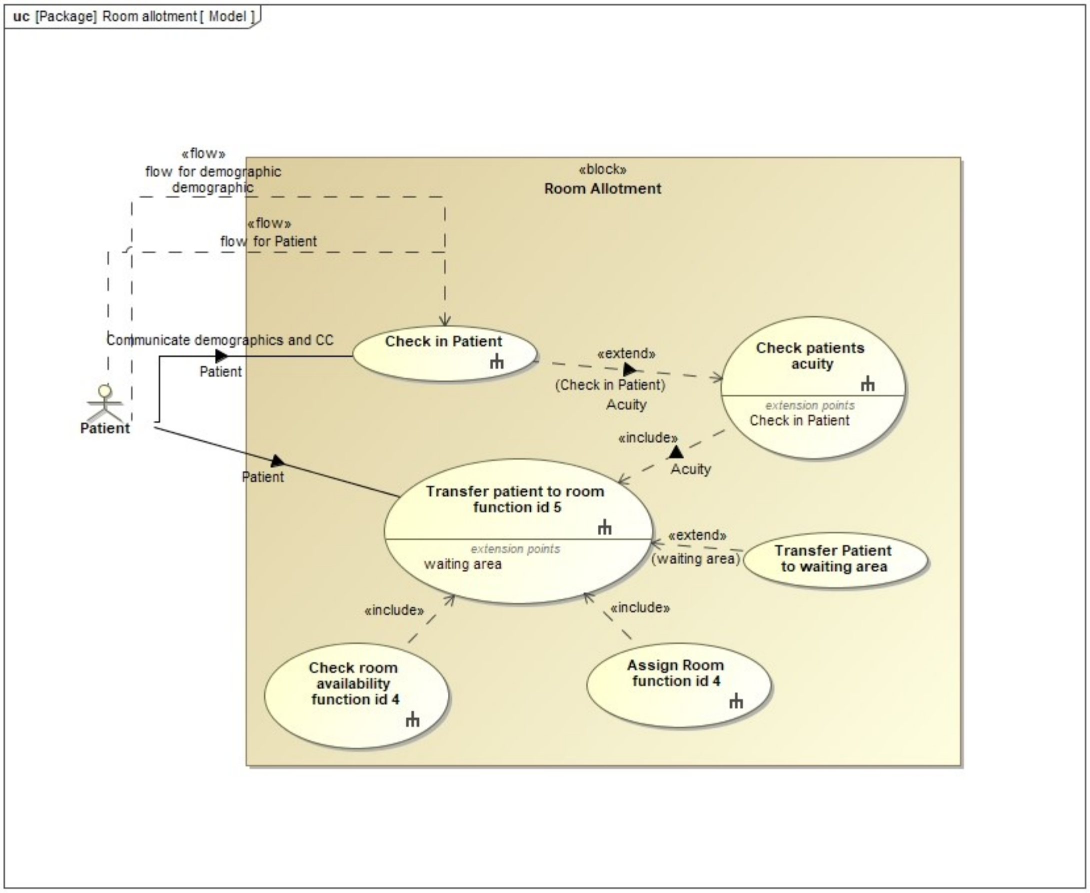

Figure 3.18. Use Case Diagram for Room Allotment

The association between use case and the actor shows that the actor is physically communicating with the system. The dashed lines show the flow of information between the actor and use cases.

D) The Overall 'Door to Room' Use Case

The overall use case of the Door to Room section of the Emergency Department is shown in 3.19. This is the combination of all the use cases as mentioned above that shows the overall behavior of registration, room allotment and patient profile generation. It shows the flow of patient from one use case to another as well. The behavior of the whole system can be understood from this use case diagram, the analyst carries out certain functions to generate the patient profile and assigns it to the patient. The patient with a profile is then registered in ED by the registered nurse and then sent for room allotment where it is sent to the room or to the waiting area depending on the availability of rooms.

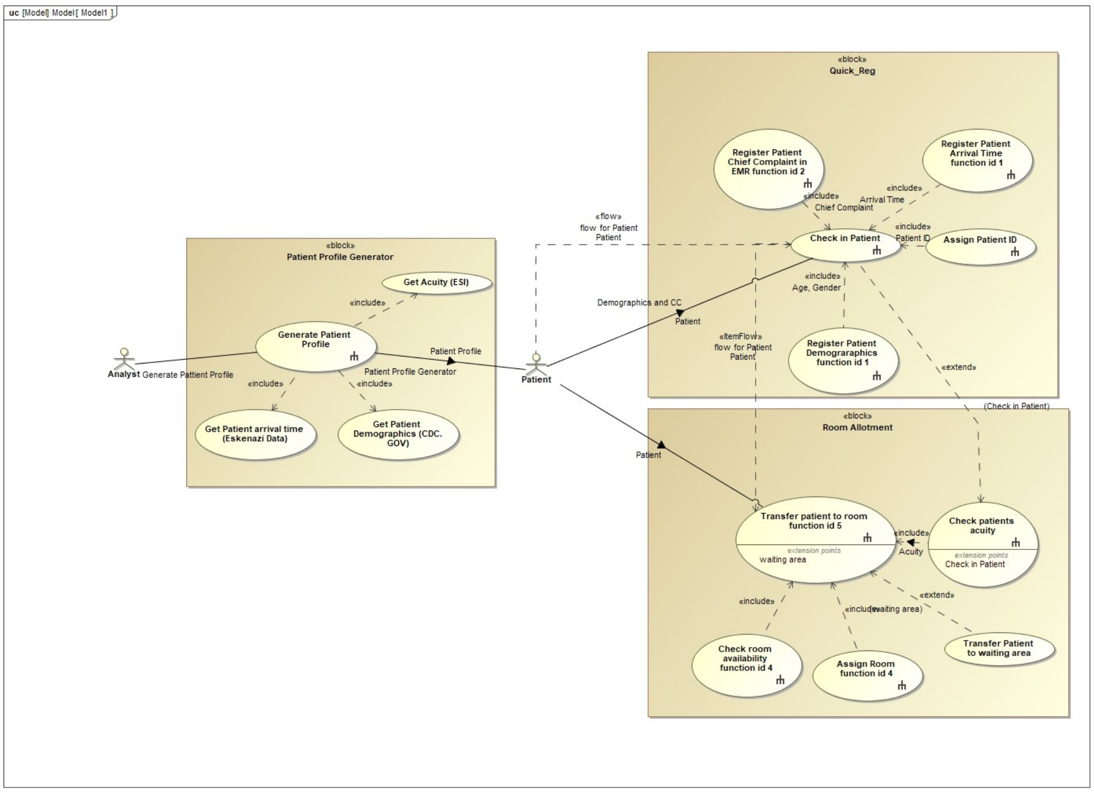

Figure 3.19. Overall Use Case of Emergency Department 'Door to Room'

3.7.2 State Machine and Activity Diagram

A State Machine Diagram is used to model the behavior of a single object, specifying the sequence of events that an object goes throughout its lifetime. State machine diagram has state actions inside it. It defines the behavior of the object on entering the state, exiting the state and while in the state. These behaviors are generally defined by nested state machine diagrams or activity diagrams. State Machine Diagrams for the three main blocks i.e. Patient, Quick Registration and Room Allotment were created. This section explains about these STMs and the corresponding state actions inside them. STMs have been defined as classifier behavior of the corresponding blocks and the same behavior defined in each base use case shown in the earlier section.

A) Behavior Model for Patient Profile Generation

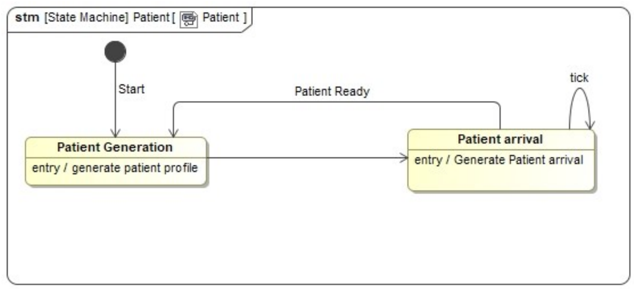

Figure 3.20. STM for Patient Profile Generation

The STM has three states, namely initial, Patient Generation and Patient arrival. Initial state is shown by a solid black dot. The arrow lines are used to show the transition between the states. The STM is by default in the initial state and the transition to the next state occurs after the start signal is sent. Patient generation state shows that the model is generating the patient profile. The actions executed in order to generate the patient profile are shown by an activity diagram defined as the entry behavior or state action of the state. The behavior of generate patient profile is shown below in figure 3.7.6. The activity diagram for generate patient profile starts with checking the iteration number of the loop. As the whole process door to room runs in a loop, it is necessary to record the number of iterations. The iteration number is stored in a property of the block Patient.

As the record of number of iterations is necessary for the model co-simulation and not for the stakeholders, this property's visibility is set to private in order to avoid unnecessary complications. The default value of the number of iterations is set as zero. This value is read with every iteration by read structural feature value action and increased by 1 by opaque action. The opaque action has feature called specification which allows to specify and execute mathematical expressions and algorithms in different programming languages. This value is then added to the property 'itr' by using add structural feature value action. This value is also transferred to action ': Generate PA' using input/output pins and object flow. Pins store values of generated from the actions momentarily, while the object flow is type of connector between two pins to carry the value. ':Generate PA' action has been defined with an opaque behavior as its base behavior. Opaque behavior is implementation specific and allows to execute specified expression/algorithm in the selected language. In this opaque behavior, language used is MATLAB. Inputs and outputs in opaque behavior are defined by creating parameters. ': Generate PA' has been implemented with opaque behavior 'Generate Patient' whose body is an algorithm to import patient data from excel and the language used is MATLAB.

The data is imported based on the iteration number, hence a parameter with direction 'in' has been defined as 'itr'. The output is patient profile i.e. Age and Gender, this is defined by parameters with 'out' direction. The parameter duration gives the time duration between arrivals of successive patients and stores it in a property of the block patient. The opaque behavior has other parameters outside the scope of this state and are used further when required. Read Self Action called 'self-imported' is used the read the context of add and read structural feature value. Once the Patient Generation is complete the STM transitions to Patient arrival state. Patient arrival starts with reading the value of duration using read self-action, this value is transferred to an opaque action. The opaque action pauses the execution of activity for the value of duration. Once the duration is complete the next action is executed using the control flow, the next action is Export door time to excel. This action has an opaque behavior which reads the current time and writes it to excel file. This time recorded is the door time of the patient or the time at which the patient arrives at the hospital, besides transferring this to excel file, door time is also added as a structural feature of the block patient. After execution of this action, control flow is used to send a signal to start room allotment, and another signal is sent to generate

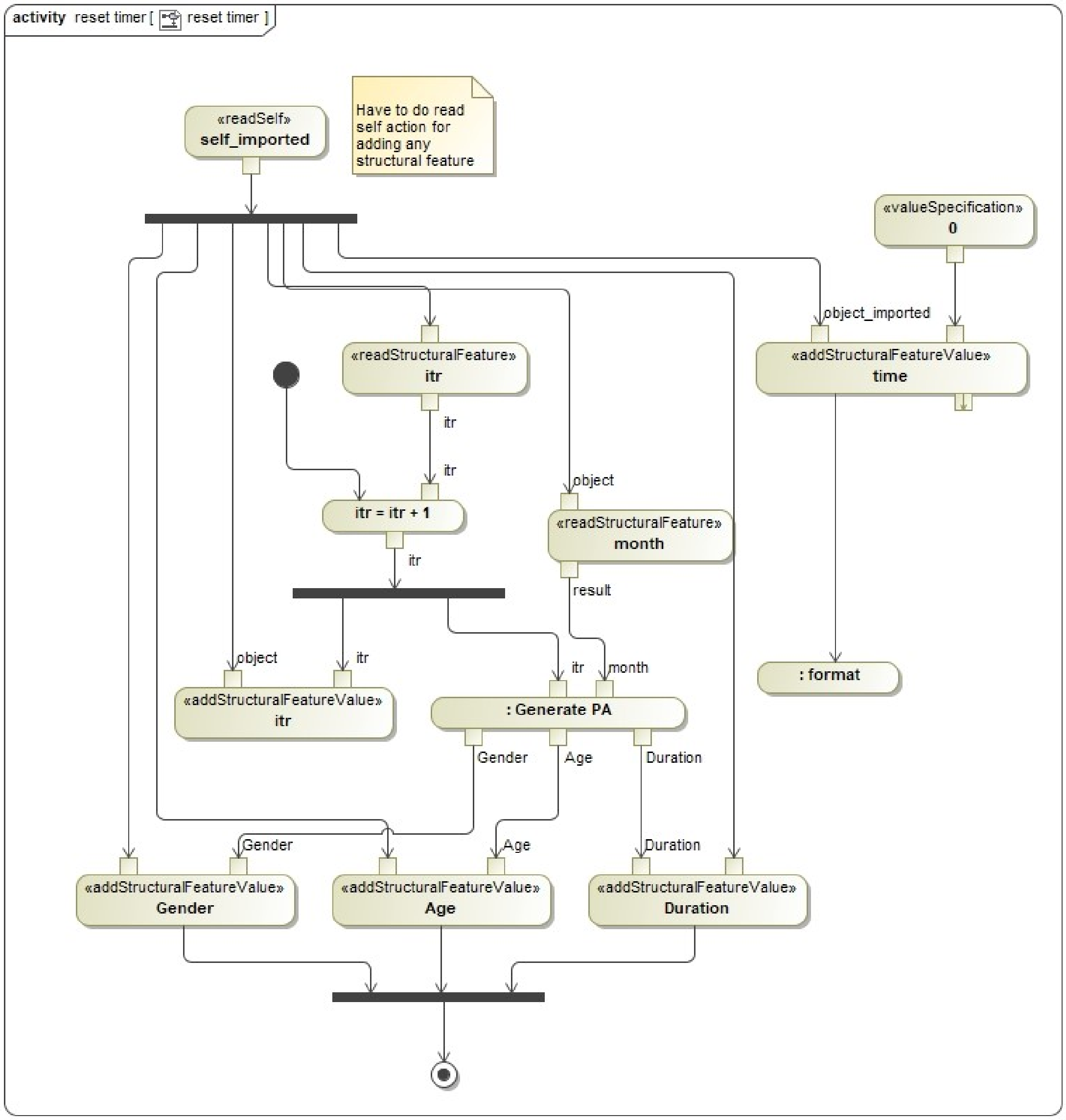

Figure 3.21. Activity Diagram for Entry/Generate Patient Profile

the next patient. These signals are used to trigger the transitions of the states. As shown in figure 3.7.7, signal patient is sent to the registration context using port named 'Patient'. And the signal patient ready is used to trigger the transition of patient generation state from patient arrival state. Again, read self-action is used to read the context of these signals.

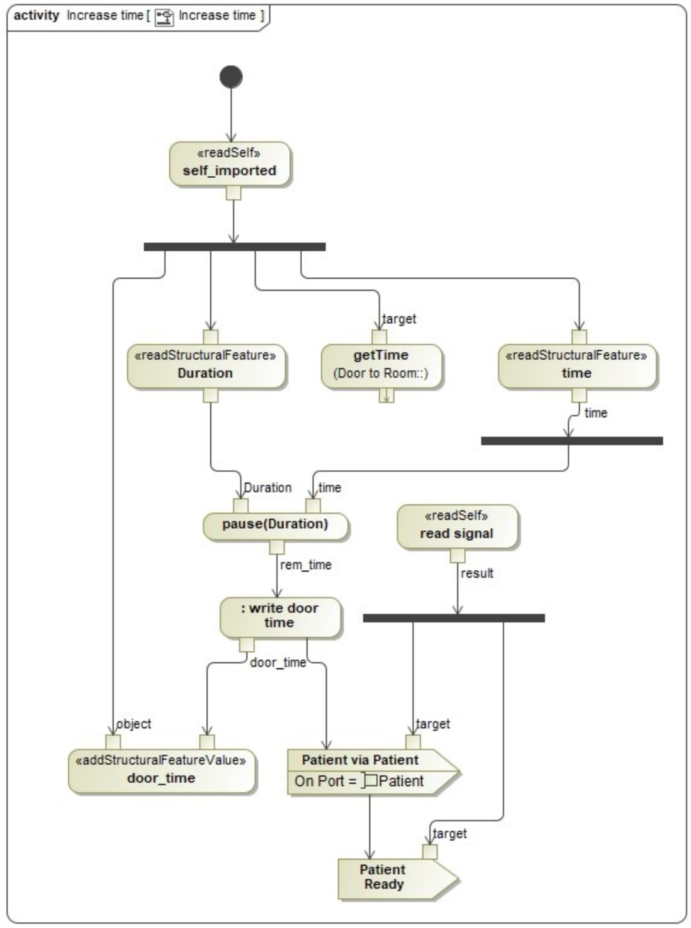

Figure 3.22. Activity Diagram for Entry/Generate Patient Arrival

B) Behavior Model Quick Registration of the Patient

The state machine diagram shown above gives the states that any registration resource (registered nurse) goes through in the working hours. It starts with getting available resource from the resource log. A resource log is a table which has resource id, shift and function id as its columns. It was created by the author himself due to unavailability of real data from the hospital. The resource id is any four digit number generated randomly using MATLAB. A day was divided in 4 parts, each part of 6 hours to depict the shift that each resource is assigned to. And function id to give an idea as to what functions the resource can perform. Function id assigned to each resource can be checked in the use case section. The state open registration resources is the first state of quick registration and is executed irrespective of patient arrival. The state has an entry behavior entry/Open Registration Resources to select a resource (registered nurse) from the excel file (resource log) based on the current real time, shift and the function id.

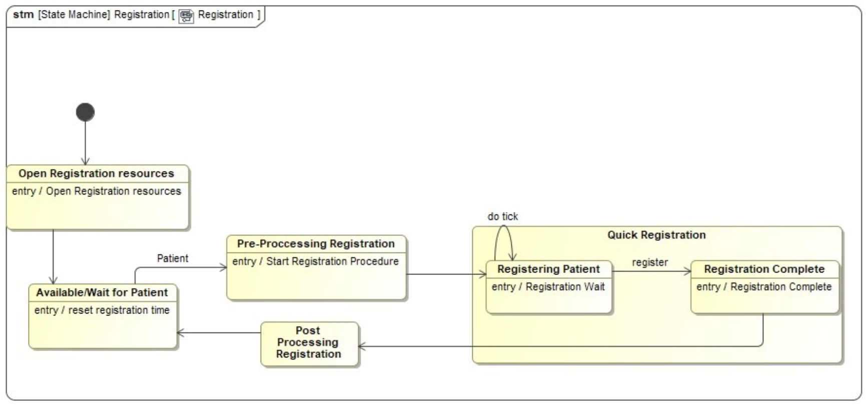

Figure 3.23. State Machine for Quick Registration of the Patient

The activity diagram starts with the action Get available resources at current time. This action has an opaque behavior whose body is a MATLAB code. It imports the resource schedule from MS Excel and assigns one resource to the registration booth. It selects this resources based on the current real time vs the shift assigned, function id and availability. There might be multiple resources available at that particular time, these are stored as property of the block registration called as 'Reg Resources'. If there are multiple available resources, one resource or the first resource is selected and is stored as the 'selected reg resource' in the context of the block registration by add structural feature value action. The availability of the selected resource is then updated in the excel file by the action Update Db for resource, which has an opaque behavior to update the availability in the excel file using MATLAB script.

Once a resource is selected, the STM transitions to Available/ Wait for the Patient. It is a state of the resource where it is sitting idle and waiting for the patient to arrive. Once the signal patient is sent by the behavior of patient generation from the port patient, the transition of Available/ Wait for the Patient to Pre-processing registration start. It is the time spent by the registered nurse before actually starting to register the patient. The state actions for this state are mainly defined for analyses purpose. It has an activity diagram as shown in 3.24, it mainly performs the actions to record the start of registration and calculate the wait time. Wait time is the time required to quick register any patient. Wait time is 5-8 minutes. It is calculated by the action get wait time for registration, which has opaque behavior. This opaque behavior has a MATLAB script as its body. The MATLAB script generates a normal distribution using multivariate random numbers between the intervals 5 to 8. And one number is picked at random from the sample size of these numbers. This selected number is the wait time for registering the patient and stored in the context as a property by add structural feature value action.

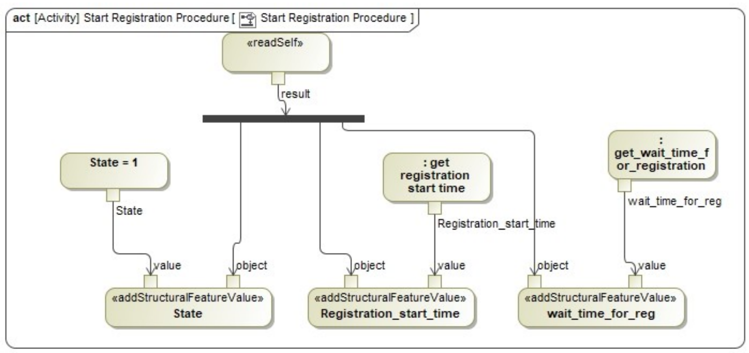

Figure 3.24. Activity Diagram for Entry/ Start Registration Procedure

The registration start time is recorded as the current time by action get registration start time. It has an opaque behavior to get the current time by MATLAB function. And is stored as the property 'Registartion start time'. State = 1 is used to show that the state of the resource has changed. This is used in the timeline chart and is explained in the results chapter. Once the activity is complete the STM transitions to state Quick Registration. Quick registration is a composite state which has two sub-states. The STM transitions to the first sub-machine state i.e. 'Registration wait'. Registration wait is a state which transitions to itself until the wait time is complete. The activity diagram for this is as shown in 3.25.

It shows that the registration wait time is read by read structural feature value action. 'time reg' is a property of the context whose value is 0. This value is increased by 1 for every iteration until its value is greater than or equal to the value of wait time

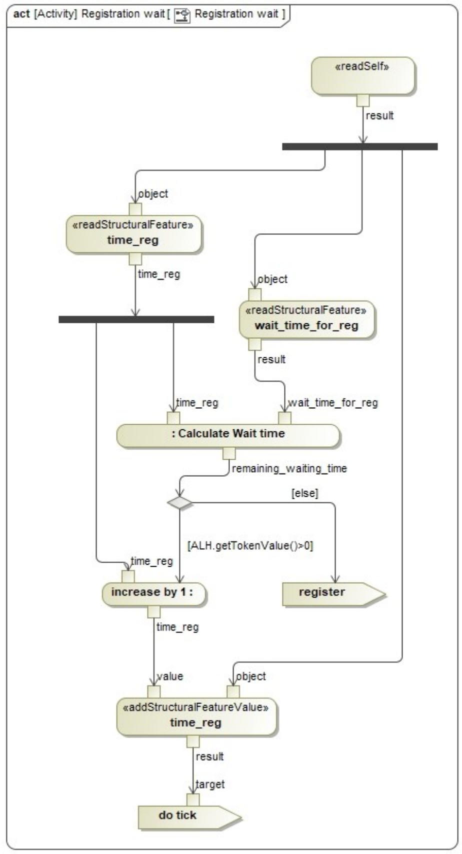

Figure 3.25. Activity Diagram for Entry/ Registration Wait

for registration. A decision node has been used to check the value. The two forks of decision nodes have guards to check the value of 'time reg'. 'ALHgetTokenValue()' is a guard used to get the last generated value. If the difference between 'time reg' and 'wait time reg' is greater than zero then the activity sends a signal 'tick' which triggers the transition of the state to self. Once the value is equal to zero, a signal 'register' is sent to trigger the transition to next state.

The next state is 'Registration Complete'. In this state the patient is registered in the EMR (In this context EMR = MS Excel) and assigned with a patient id. Figure 3.7.13. It starts with reading the number of iterations. The default value of number of iterations is 0, and it is increased by one for every loop. The opaque behavior for Get data from Patient selects the patient data based on the number of iterations. This data is transferred to another opaque behavior 'Transfer data' which transfers the data to MS Excel. A patient id is assigned to the patient which is a four digit number selected on the basis on the number of iterations and is updated as a structural feature of the context.

After the data is transferred, the registration end time is recorded and updated as the property of the block registration. After adding the structural feature, control flow is used to send a signal to start room allotment procedure. This signal is sent to the port registered patient which is connected by a binding connector to the block room allotment as shown in the IBD of Door to Room. Once the signal is consumed, STM triggers the transition to post-processing registration and the patient then transitions back to Available/wait for patient, where the registration clock is set to the default value i.e. 0. It remains in this state until the next patient or until the shift of the registered nurse gets over. The transition to open registration resources is triggered after 3600 seconds or 60 minutes. (In this context the 1s=1min). This trigger leads to selecting the next available resource from the resource log.

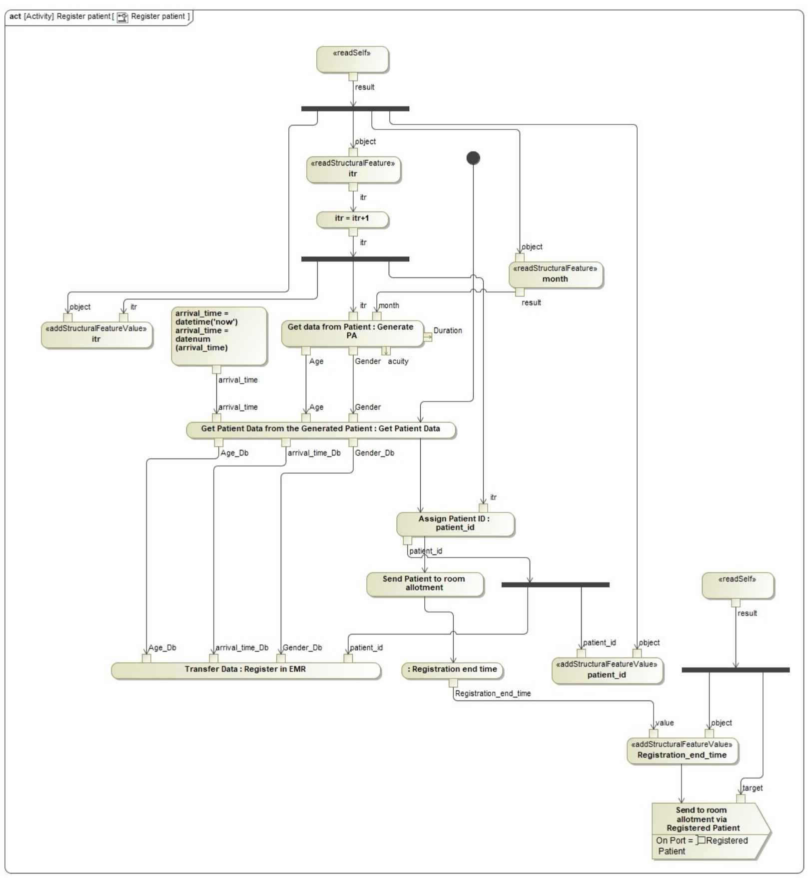

Figure 3.26. Activity Diagram for Entry/ Registration complete

C) Behavior for Room Allotment

A state machine diagram shows the states of the CareTech throughout its working hours. It starts with getting the available CareTech at the current real time on the basis of its shift, function id and availability. This state follows the same procedure as described in the Quick Registration State Machine for the state Open registration resources to select the room allotment resource or CareTech. Once the CareTech is assigned it waits for the patient. After the patient comes in for room allotment, the resource finds the available rooms according to its acuity level. And transfers the patient to either the room or to the waiting area on the basis of room availability. 3.28

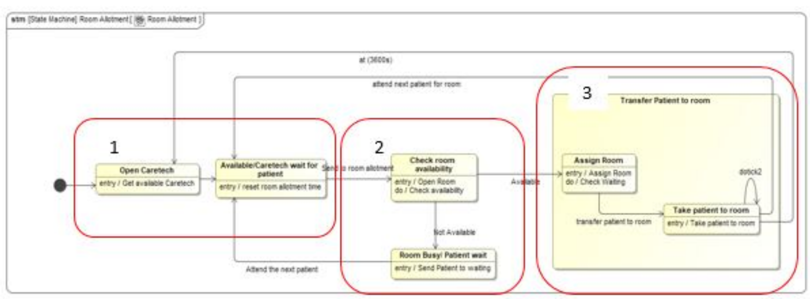

Figure 3.27. State Machine Diagram for Room Allotment

shows the first section of figure 3.27. The two states shown in the figure have same behavior as the first two states of Quick Registration STM. State action entry/Get available CareTech is an activity diagram which imports the CareTech data from the resource log and selects the resources that are available at the current real time and updates their availability in the log. The resource ids of the available CareTechs are updated as the property of the context as shown in 3.29. After selection of the CareTech, the STM transitions to Available/CareTech wait for the patient. This is a state where the CareTech is waiting for the patient to get registered by the registration staff and sent to the room allotment section. For co-simulation and result post-processing purpose, this state has certain actions such as 'room allotment time' to record the time required to record the time spent by the CareTech on each patient.

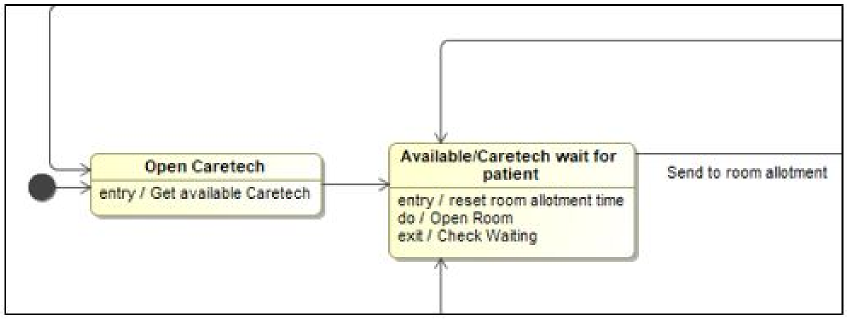

Figure 3.28. Magnified View of Section 1 of Room Allotment STM

And 'State' to plot the data on a timeline chart. The purpose of these actions will be clearer in the subsequent chapter in the co-simulation section. The state Available

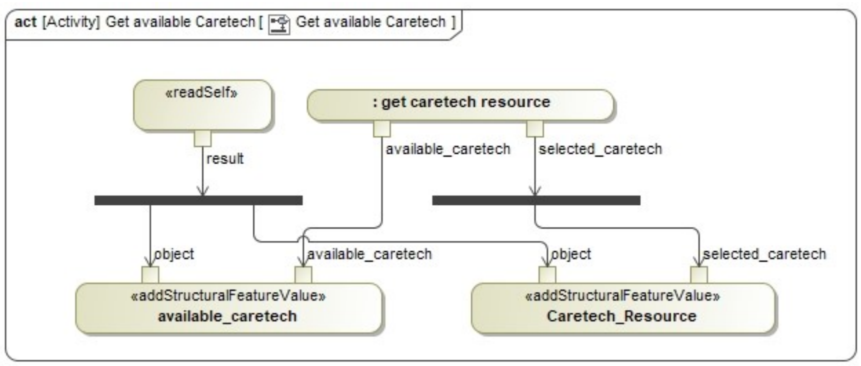

Figure 3.29. Activity Diagram for Entry/Get Available CareTech

CareTech/Wait for patient has three state actions, entry/reset room allotment time, do/open room and exit/ Check Waiting. The entry behavior is executed first. The state action do/open room is shown in 3.31. Open room is a behavior of the CareTech for every loop, it shows the CareTech checks for rooms that are to be opened after discharge of previous patients. And the exit behavior shows that if the rooms are open then the CareTech will transfer the patient from waiting area to the room. Similar to the CareTech log, an emergency room log was created by the author in MS Excel. It includes the room numbers from different acuity units and their corresponding availability. Whenever a room is assigned to a patient, the time of room allotment is

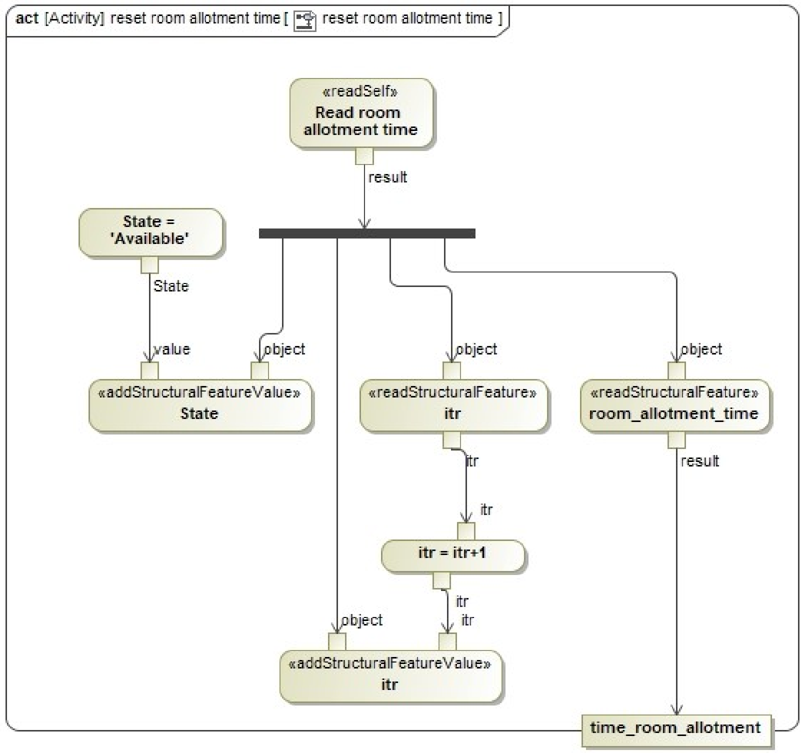

Figure 3.30. Activity Diagram for Entry/Reset Room Allotment Time

recorded. Patient from different acuity levels occupy the room for different amount of time. Based on the average time of occupancy by patient of different acuity levels a MATLAB function was created to open the rooms of different acuity levels at different times. So whenever a patient occupies the room, time of room allotment is

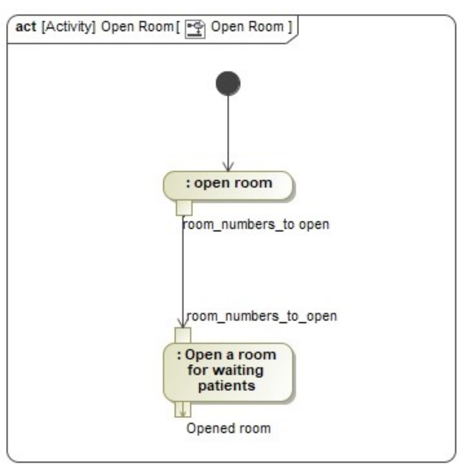

Figure 3.31. Activity Diagram for Do/Open Room

recorded. And based on the average time of occupancy, the room is made available after the completion of that time. The numbers for the rooms that need to be opened are transferred to the action 'Open a room for waiting patients', this action has an opaque behavior in language MATLAB which updates the room log for the given room numbers. There can be zero room numbers as well, whenever there are no rooms to be opened the activity will terminate and will execute the next state action. This step is not required for the first iteration of the co-simulation but is necessary for the subsequent iterations. Once the room is opened or once the activity Open room is executed, the state executes its exit behavior, exit/Check waiting is shown in 3.32. The activity shows the CareTech reads the waiting database to check the patients in the waiting area. If there are any patents in the waiting area and the corresponding a room is available in the corresponding acuity unit then the patient is transferred to the room, otherwise the activity is terminated. When the patient is transferred to the room, the corresponding rooms availability is updated in the room database (here room excel file). Simultaneously a separate excel file is updated which records the time at which the patient is transferred to the room. This is done by an opaque behavior specified by a MATLAB function in the actions 'assign room' and ': Update room Db'. Once these state actions are completed the STM Room allotment remains in this state until it receives signal 'send to room allotment'. The second section consist of checking the availability of the room. Depending upon the availability the patient is transferred either to the waiting area or to the room. The magnified view of part 2 of 3.27 is shown in 3.28. Once the signal from port registered patient is received the transition from 'Available/Wait for patient' to 'Check room availability' is triggered. Once the rooms are opened, the state action entry/check room availability is performed. This activity is necessary as the rooms are opened in the previous state may not be necessarily the rooms with the acuity same as that of the incoming patient. That means opening of the occupied rooms is not dependent on patient arrival. The following figure shows the activity for state action entry/check room availability. The activity starts with checking the patient's acuity level. Based

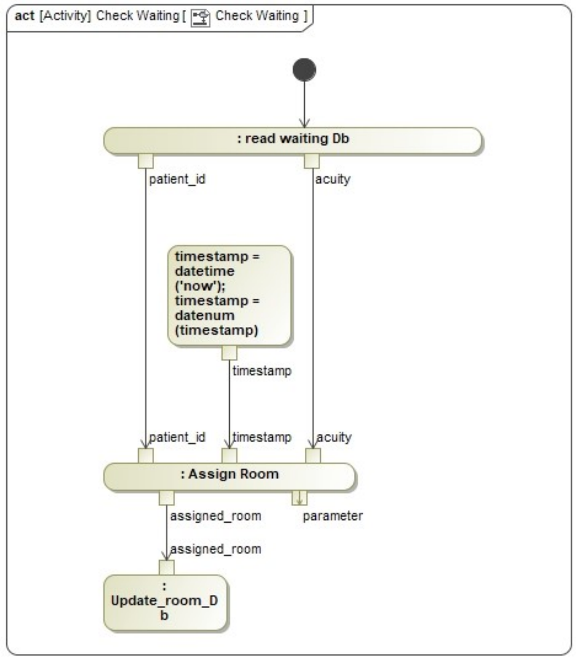

Figure 3.32. Activity Diagram for Exit/Check waiting

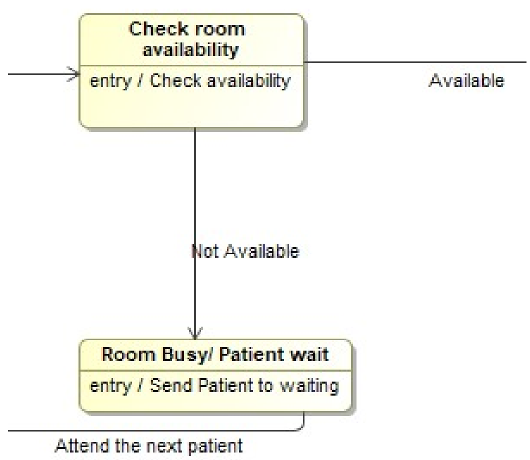

Figure 3.33. Magnified View of Section 2 of Room Allotment STM

on the patients acuity level the room availability is checked using the opaque behavior 'Get Room database'. This opaque behavior has a defined MATLAB function which imports the room availability data from excel and checks if the corresponding acuity level room is available. For example: If the incoming patient has been assigned with

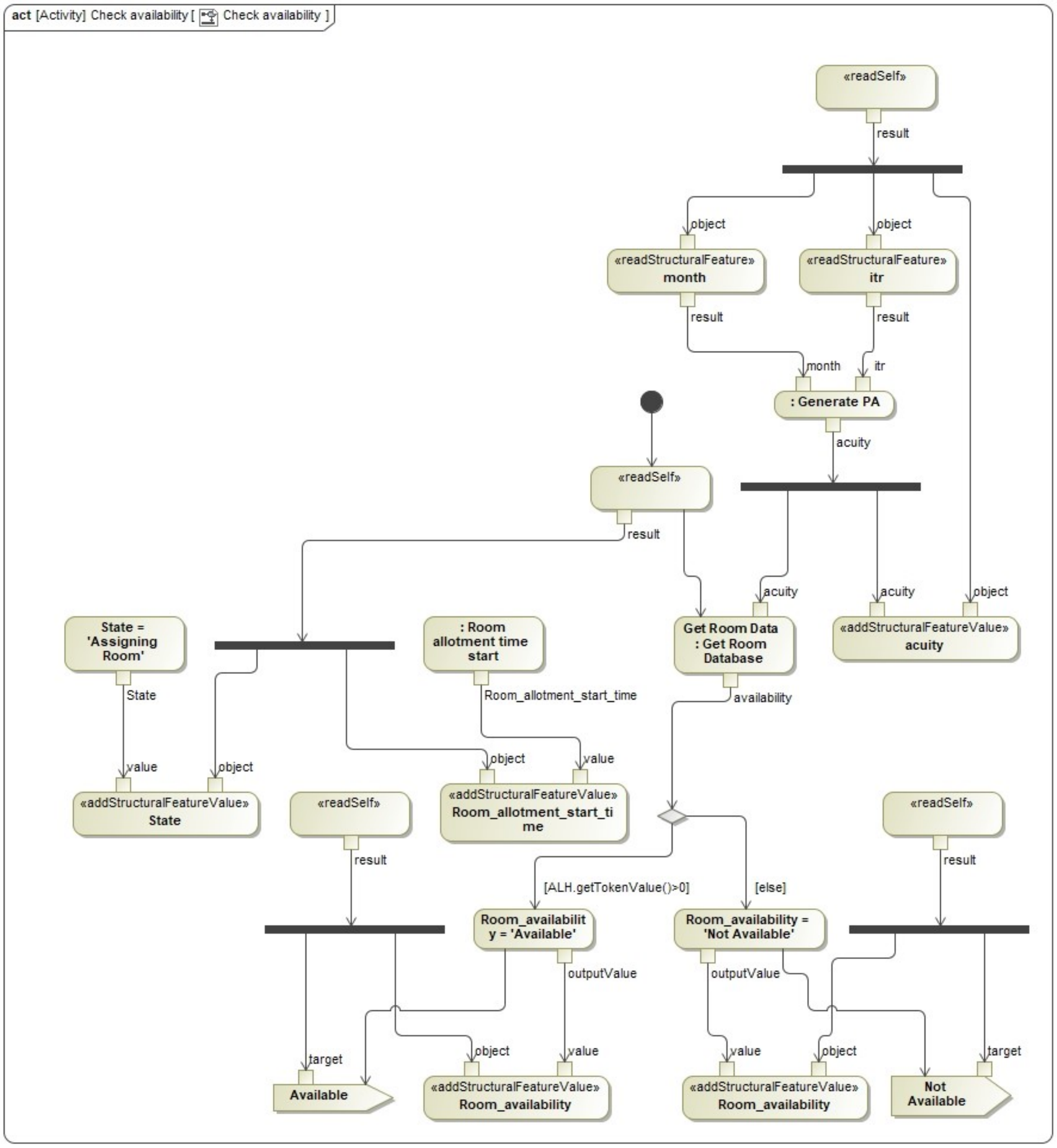

Figure 3.34. Activity Diagram for Entry/ Check Room Availability

Low Acuity, the MATLAB function will check the imported data for rooms available in the Low Acuity unit. Simultaneously, the room allotment start time is recorded by add structural feature value action and the state of CareTech is updated from 'available' to 'assigning room'. Based on the availability the activity sends either of the two signals 'Available' or 'Not available'. This selection is done by a decision node. The opaque behavior gives a binary output. If a room is available then it gives an output of 1 and if not 0. An ALH API is used on the guard of control flows to check the latest value of availability and if the value is 1 then it sends signal 'Available' else it send 'Not Available'. If the room is not available then the Patient is sent to the waiting area. The signal 'Not Available' triggers the transition to state 'Room Busy/Patient Wait'. The CareTech transfers the patient to waiting area and then attends the next patient. This activity of the CareTech is shown by the state action entry/Send Patient to Waiting.

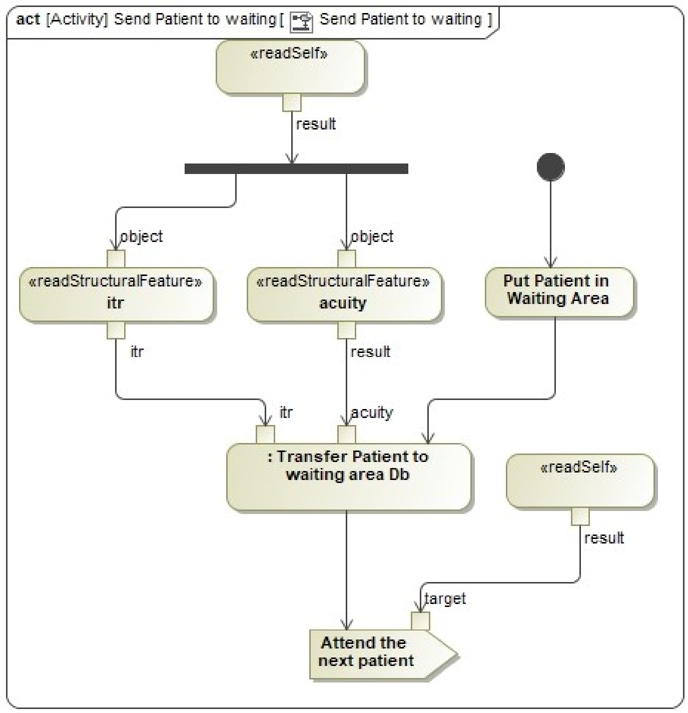

Figure 3.35. Activity Diagram for Entry/Send Patient to Waiting

The activity diagram shows that the patient is transferred to the waiting area and the patient data is transferred to database, in this context an excel file. This action is done by an opaque behavior, which has been defined with a MATLAB function as the specification to transfer the patient data [acuity, patient id and timestamp] to an excel file. The control flow then sends a signal 'Attend next patient' which triggers the transition to state 'Available/CareTech wait for patient'. As shown in the 3.34, if signal 'Available' was sent then, the transition to the state 'Assign room' would have been triggered. Section 3 of figure 3.27 shows the corresponding states executed if signal 'Available' is consumed.3.36 shows the magnified view of section 3. The state

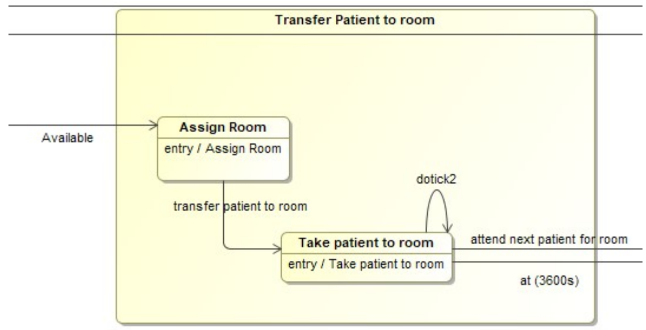

Figure 3.36. Magnified View for Section 3 of Room Allotment STM

assign room has an entry behavior entry/Assign Room. This behavior shows that the CareTech assigns a room to the patient based on the acuity of the patient and updates the Emergency room log. The activity assign room is shown in 3.37, it shows that the CareTech checks patient's acuity, gets the available room from the emergency room log, assigns it to the patient. ': Assign room' has an opaque behavior that updates an excel file to with patient id and the assigned room number. The activity diagram then executes the action update room database which also has an opaque behavior specified by a MATLAB function which updates the room availability in the excel file. After the room excel file is updated the activity terminates by sending a signal 'transfer patient to room'. It triggers the next state 'Take patient to room'. The activity diagram shown in figure 3.38 show the entry behavior of the state 'Take Patient to room'. It generally takes 6-7 minutes to assign room to patient and take him to the room. The opaque action 'room allotment time start' records the time at which the patient transfer initiates. ': get wait time for room allotment' has been specified with an opaque behavior which is a MATLAB function which generates the wait time from a normal distribution of wait times. This wait time is transferred to an opaque action which pauses the activity till the wait time is complete. After that the patient is transferred to the room, this time is recorded as the time at which the patient goes in the room ('room time'). After the patient is transferred to the room, the room allotment time is recorded by add structural feature value action, the input to this action is given by an opaque behavior action which has a MATLAB function

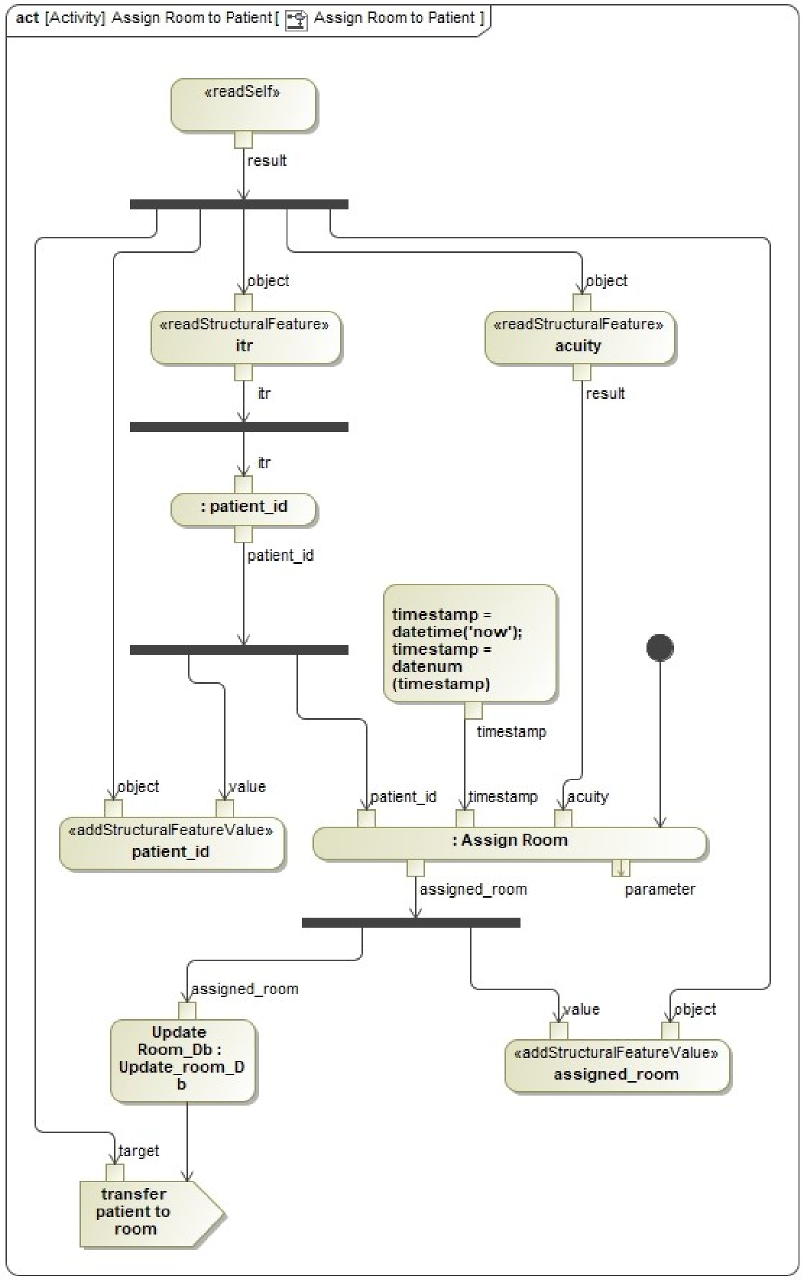

Figure 3.37. Activity Diagram for Entry/ Assign Room

to get the current time. After this control flow is used to send signal to trigger the transition to 'Available/ CareTech wait for patient'. The 3.39 shows the overall view of the Door to Room behavior. It shows the behavior of each block and the transfer of patient from one block to another. This model was further used for co-simulation purpose.

3.8 Verification of 'Door to Room' Model

Verification is the process of checking if the model is built right. It emphasizes on verifying if all the requirements of the model are met in the right manner. Verification of the 'Door to Room' model was done using the Requirements Verification Matrix in Cameo Systems Modeler . Requirements Verification matrix is a dependency matrix which allows to analyze, create, or modify verification relations between requirements and named elements that can determine whether the systems fulfill the requirements. 3.40 shows Requirement Verification Matrix. It shows the requirements and corresponding elements of the model that verify the requirement. The method of verification of the requirements is by inspection. Inspection can be done by inspecting the console window and the variables window while the model is simulating. The console window provides with every output generated. There is an option of selecting the MATLAB language for the console window. So if the outputs from MATLAB are not suppressed, they will show up on the console window.

The interface of Cameo-MATLAB-MS Excel can be verified by inspecting the patient data appearing on the console window as shown in figure **??**

The variable pane shows the runtime values generated from the co-simulation. As shown in the requirements verification matrix, many of the requirements are verified by properties of the respective context. 3.41 shows the variable pan that shows the properties for the context patient and Registrations. From the variables pane it can be seen that requirements 1.1.3, PPG 2.0.1, Quick reg 2.0, Quick reg 2.4 are getting verified. Besides the Requirements Verification Matrix, a relation map was also developed to show the relationship between requirements and the element verifying it. Relation map is shown in [**?**]. Relation map allows to visually analyze amongst multiple layers of abstraction. All these diagrams help to analyze the working of the system in compliance with the defined requirements.

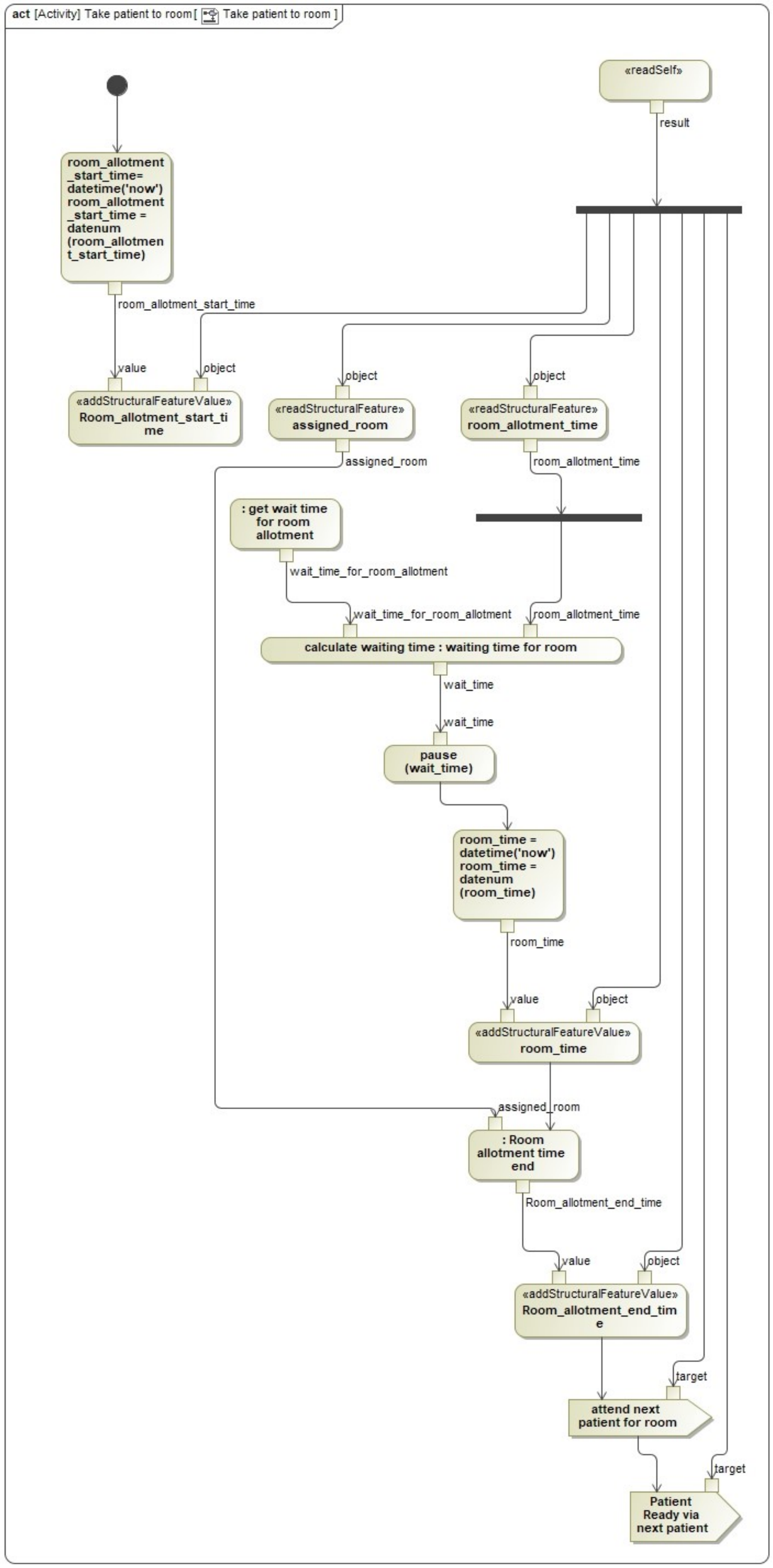

Figure 3.38. Activity Diagram for Entry/ Transfer Patient to Room

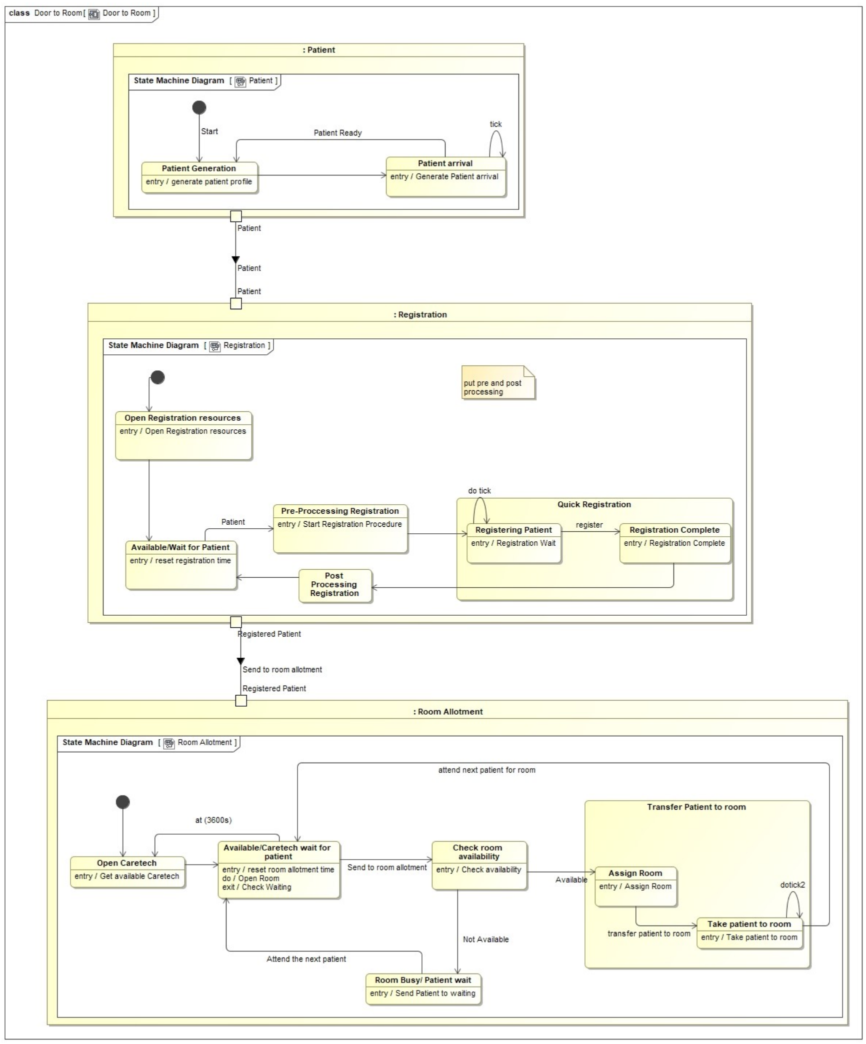

Figure 3.39. Overall View of the 'Door to Room' Behavior Model

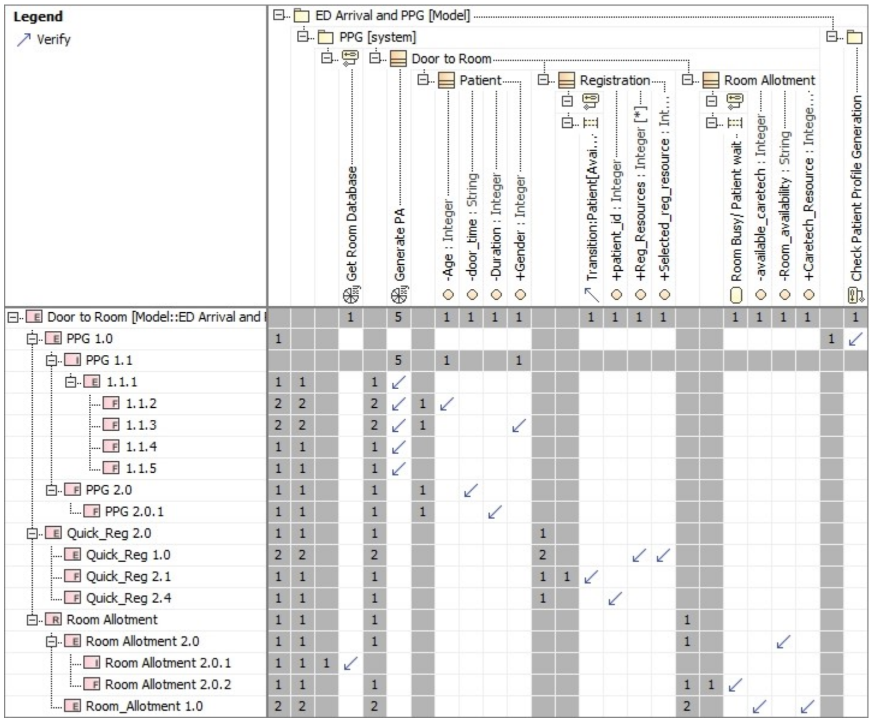

Figure 3.40. Requirements Verification Matrix

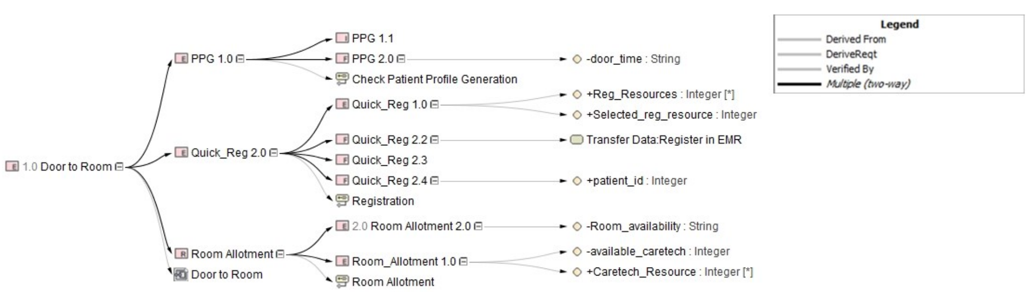

Figure 3.41. Requirements Relation Map

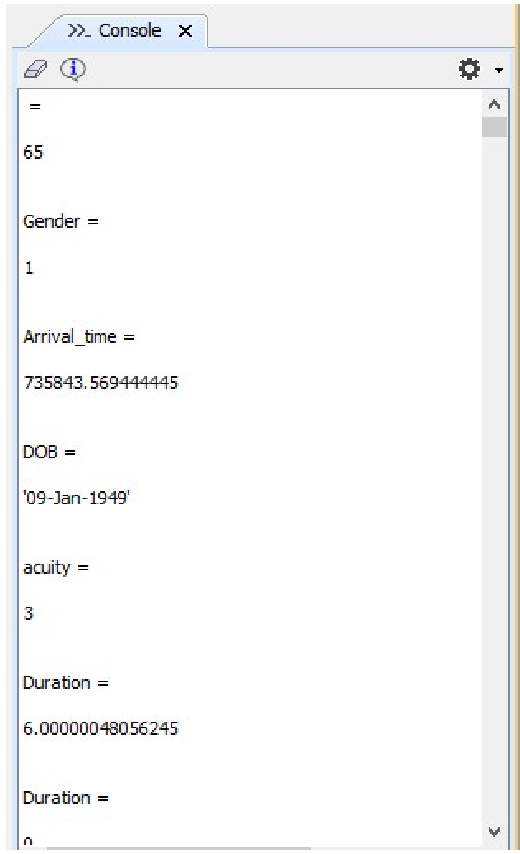

Figure 3.42. Verification of Requirement id 1.1.1

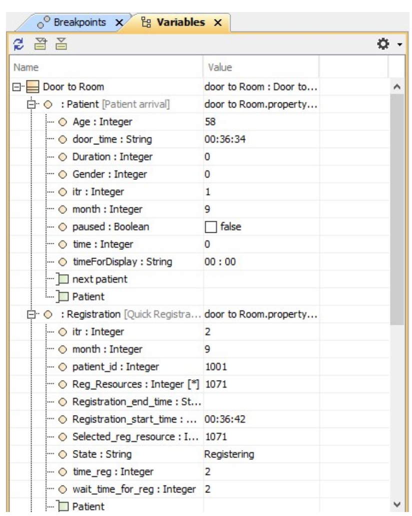

Figure 3.43. Variables Window for Inspection of Requirements Verification

4. RESULTS AND DISCUSSIONS

This chapter talks about the verification of model, co-simulation of model using a case study and discussion on the co-simulation results. The verification was done using Cameo, it gives an ability to verify the specified requirements using requirement verification matrix. The model shown in figure 3.3.26 was used to simulate a case study. Further this chapter talks about the Graphic User Interface used to simulate the model. The model was simulated for multiple case studies, one of such case study is described below. The model was simulated for twelve hours of 1st November. There were a total of 140 patients that arrived during this time interval. Following parameters were recorded by the co-simulation model: 1. Door to Room Time 2.Time spent at registration 3. Time spent at room allotment. The Door to room time for each patient arrived is shown in figure 4.1. The X axis is the patients arrived and the Y axis is time in minutes. The plot shows bar graph for Door to room time for each patient. 4.2 shows the Door to Room time for each patient. The X axis is hour of the day and Y axis is the time required for the patient to get to the Room from Door. The plot shows poisons distribution as shown and explained in section 2.2.1. It shows that Door to Room time for patient when arrived at between 10:00 Hrs to 20:00 Hrs is more than for patients arrived at other time of day. This is because, the historic data shows that at these hours of the day, maximum number of patients arrive at the emergency department which leads to queuing of the patients. 4.3 shows a 3D plot of Door to Room time vs Age vs Hour of Day. It is observed from this figure that the Door to Room time for patients of age group 60 to 80 is the most ranging from 100 to 120 minutes as most of these patients arrive at busy hours.

Besides this plot, a time series chart was used to monitor the resource states throughout the co-simulation. Time series charts is for both registration and room allotment were plotted in cameo. These charts are real-time, the X axis is the times-

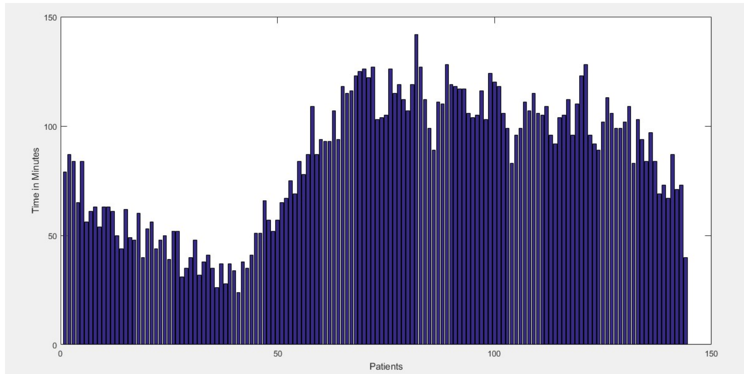

Figure 4.1. Plot for Door to Room Time for Each Patient

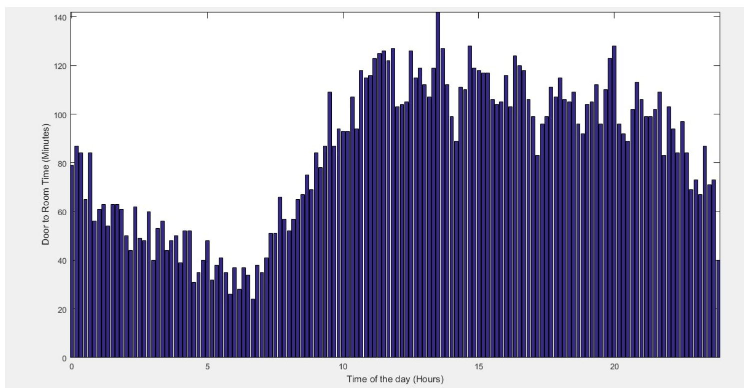

Figure 4.2. Plot for Door to Room Time vs Time of Day

tamp in milliseconds and Y axis is the resource state. In the following chart state of registration resource is plotted in real time, The Y axis is the state of resource where 1=Available and 0=Not Available. The time for which the Y axis value is 0 is the time for which patient is getting registered.

Besides Time series charts, the co-simulation data was also collected using co-simulation logs and the exported to *.csv file. The data that was recorded was 1. Registration Start Time 2. Room Allotment Start Time and 3. Room Allotment End time 4. Patient ID. This data gives the exact information about the time spent by the patient in each state as well as the time spent by the resource on each patient, idle time of the resource etc. This data can be further made more useful on post

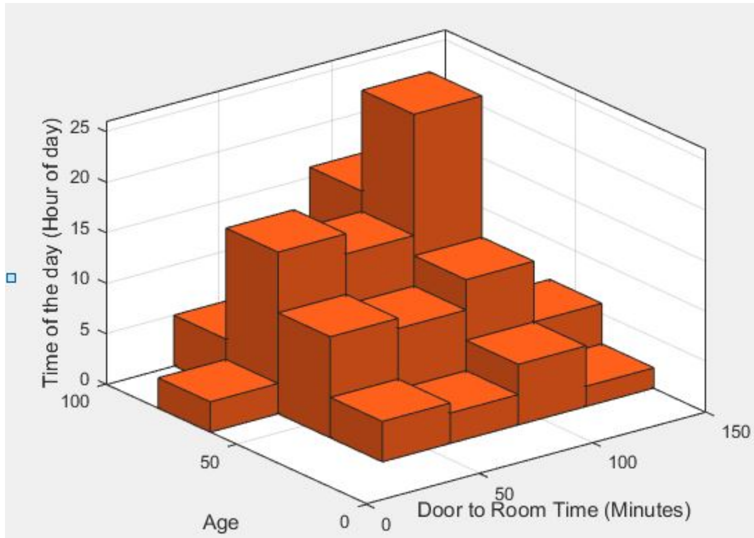

Figure 4.3. Plot for Door to Room Time vs Age vs Hour of Day

Figure 4.4. Time Series Chart

processing. 4.4 shows the values that were recorded during the co-simulation. These values are stored as instance slot values in the form of tables in Cameo Systems Modeler . These tables can also be imported into Microsoft Excel for better post processing. 4.5 shows one of such tables imported in Microsoft Excel which shows the starting and ending time for registration and room allotment. This table provides information on time spent by human resources on patients at the quick registration desk and at the room allotment section.

+patient_id : Integer	1000	- Registration_start_time : String	16:59:09	- Room_allotment_start_time : String	4:58:21 PM	- Room_allotment_end_time : String	16:58:52
+patient_id : Integer	1001	- Registration_start_time : String	17:00:21	- Room_allotment_start_time : String	4:59:28 PM	- Room_allotment_end_time : String	17:00:01
+patient_id : Integer	1002	- Registration_start_time : String	17:01:30	- Room_allotment_start_time : String	4:59:28 PM	- Room_allotment_end_time : String	17:01:12
+patient_id : Integer	1003	- Registration_start_time : String	17:02:42	- Room_allotment_start_time : String	5:00:40 PM	- Room_allotment_end_time : String	17:02:25
+patient_id : Integer	1004	- Registration_start_time : String	17:03:52	- Room_allotment_start_time : String	1+E430:E777 7:00:40	- Room_allotment_end_time : String	17:03:35
+patient_id : Integer	1005	- Registration_start_time : String	17:05:25	- Room_allotment_start_time : String	5:01:51 PM	- Room_allotment_end_time : String	17:04:43

Figure 4.5. Table of Time Spent by Each Resource on Each Patient

A graphic user interface was developed for the model in cameo. Graphic user interface allows the real time monitoring of the resource states and the time spent by patient in each state. It also shows the available resources and the assigned resource. 4.6 shows the GUI for Quick registration. Similar GUI was developed for room allotment section.

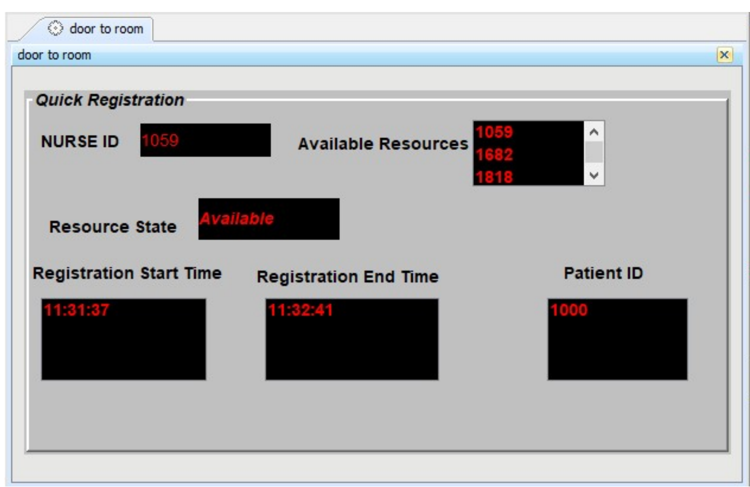

Figure 4.6. GUI for Quick Registration

5. CONCLUSIONS AND FUTURE WORK

A generic model of ED Door to Room behavior was built using a group of system-level views specifying the ED systems structure, behavior, requirements and function. MBSE approach was used to create a structural and behavioral model of the Door to Room section of a generic emergency department, simulate this model in real time to see the various states and the corresponding time each resource spends on a particular patient and the time required by the patient to get to the emergency room after arrival at the hospital. The model was simulated in Cameo Co-simulations Modeler. The co-simulation was done using the trends of patient arrival using historic data. The co-simulation calculates the patient door to room time and the time spent by the resource on each patient. This will allow the hospital administrator to utilize the resources efficiently. With co-simulation and analysis, this research shows that MBSE using SysML can be applied for co-simulation and analysis of complex systems in real time.

It is clear that the proposed work has a very few number of data which lead to less accurate results. However this model can serve as guideline to build a complete model which can be brought into practical use. To build a complete model, more data will be required. The future work consists of collecting precise data of the Door to room section. This data consist of time required by registration nurse to register patients of different acuity levels, efficiency of registration of each registration nurse, time required to register patients of different age and gender. Collection of this data will give an accurate distribution of time required to register patients with different age, gender and acuity level. Similarly data shall be collected for room allotment section. The data will consists of efficiency of each CareTech to assign room and transfer the patient. The time required by the CareTech to assign the room and to transfer the patient to corresponding room may vary with respect to the age, gender, and acuity level. Therefore, collection of this data will give a distribution of time required by the CareTech to assist patients with different profile. Also, the room occupancy time for patients with different age and gender may vary. Therefore, this data shall be collected in the future to get accurate results from the model.

This model is linear, meaning that there is one resource, one patient at any given instance. The future work will consist of making this model dynamic. The data for schedules of different resources working in the Emergency department shall be collected in the future. Using this data model can be made dynamic. In contrasts to the future work, this model also shows that simulation and analysis can be carried out it the behavioral modeling than in the parametric modeling. Doing Simulation and analysis in this phase actually gives a better overview and understanding of the system as it describes the action executed by each element of the system while achieving the system objective. As in this case, it shows the actions executed by nurse, CareTech and the patients individually, transfer of data and information between these elements and evaluation off mathematical constraints in the behavioral pillar of the framework. Whilst the previous approach shows the same by simulating the system in parametric modeling, it does not show the order in which the actions are executed. The parametric only evaluates the blocks irrespective of their order of evaluation. Hence the author believes that the optimum point for integrating the model for co-simulation is by co-simulating the behavior models rather than the parametric.

Though this approach has some shortcomings currently, they can be overcome. One of the biggest limitation of this approach is updating the block values while co-simulating the behavior model. The current model does not provide any provision for updating instances when the behavior model is executed or co-simulated with MATLAB. However, the author believes this limitation can be overcome by adding additional constraints to the action blocks. Using relevant JAVASCRIPT codes the instances could be updated real time. SysML tools (Cameo Systems Modeler) have used JAVASCRIPT as the language for software development, hence JAVASCRIPT can be used as an Application Programming Interface (API) to further extend the capabilities of SysML tool, in this case update instances and block values.

Besides these shortcoming, another factor that is altering the results from the model is the error between simulation time and the execution time. This error is not specific to the model but it depends on the operating system specifications and its capability. The author believes better specifications of the system, correct amount of Random Access Memory allocated for the SysML tool and corresponding co-simulation tool and a separate JAVA for the SysML tool can produce accurate results from the co-simulation models.

REFERENCES

REFERENCES

[1] S. Group, "2010 survey of: Hospital emergency department administrators," *Schaumacher Group*, pp. 1–2, 2010.

[2] N. Jayaprakash, R. OSullivan, T. Bey, S. S. Ahmed, and S. Lotfipour, "Crowding and delivery of healthcare in emergency departments: the european perspective," *Western Journal of Emergency Medicine*, vol. 10, no. 4, p. 233, 2009.

[3] J. S. O'Shea, "The crisis in america's emergency rooms and what can be done," *Researchgate*, pp. 1–2, 02 2018.

[4] G. Kaplan, G. Bo-Linn, P. Carayon, P. Pronovost, W. Rouse, P. Reid, and R. Saunders, "Bringing a systems approach to health," *Institute of Medicine of the National Academies*, p. 1, 2013.

[5] R. Kopach-Konrad, M. Lawley, M. Criswell, I. Hasan, S. Chakraborty, J. Pekny, and B. N. Doebbeling, "Applying systems engineering principles in improving health care delivery," *Journal of General Internal Medicine*, vol. 22, no. 3, pp. 431–437, 2007.

[6] I. S. Khayal and A. M. Farid, "The need for systems tools in the practice of clinical medicine," *Systems Engineering*, vol. 20, no. 1, pp. 3–20, 2017.

[7] SEBoK, "Incose systems engineering vision 2020 sebok," *Stevens Institute of Technology Systems Engineering Research Center*, p. 1, 2012.

[8] D. Zwemer, "Technote: Applications of mbe for healthcare," *Intercax LLC*, pp. 1–15, 2016.

[9] O. Casse, "We choose mbse: What next," *NoMagic*, pp. 2—10, 2015.

[10] S. Corns and C. Gibson, "A model-based reference architecture for medical device development," in *INCOSE International Symposium*, vol. 22, no. 1, 2012, pp. 2066–2075.

[11] A. L. Ramos, J. V. Ferreira, and J. Barceló, "Model-based systems engineering: An emerging approach for modern systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 1, pp. 101–111, 2012.

[12] A. T. Bahill and R. Botta, "Fundamental principles of good system design," *Engineering Management Journal*, vol. 20, no. 4, pp. 9–17, 2008.

[13] T. Weilkiens, "Systems engineering with sysml/uml: modeling, analysis, design," *Elsevier*, pp. 1–2, 2011.

[14] R. S. Peak, R. M. Burkhart, S. A. Friedenthal, M. W. Wilson, M. Bajaj, and I. Kim, "9.3.2 simulation-based design using sysml part 1:a parametrics primer," in *INCOSE International Symposium*, vol. 17, no. 1. Wiley Online Library, 2007, pp. 1516–1535.

[15] A. P. Sage and C. L. Lynch, "Systems integration and architecting: An overview of principles, practices, and perspectives," *Systems Engineering*, vol. 1, no. 3, pp. 176–227, 1998.

[16] D. Kaslow, L. Anderson, S. Asundi, B. Ayres, C. Iwata, B. Shiotani, and R. Thompson, "Developing a cubesat model-based system engineering (mbse) reference model-interim status," in *Aerospace Conference, 2015 IEEE*. IEEE, 2015, pp. 1–16.

[17] T. Johnson, A. Kerzhner, C. J. Paredis, and R. Burkhart, "Integrating models and simulations of continuous dynamics into sysml," *Journal of Computing and Information Science in Engineering*, vol. 12, no. 1, p. 011002, 2012.

[18] K. Azevedo, B. Bras, S. Doshi, and T. Guldberg, "Modeling sustainability of complex systems: A multi-scale framework using sysml," in *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2009, pp. 1437–1448.

[19] P. C. of Advisors on Science and Technology, "Report to the president better health care and lower costs: Accelerating improvement through systems engineering," *Executive Office of the President*, pp. 6–48, 2014.

[20] O. G. Batarseh, E. J. Goldlust, and T. E. Day, "Sysml for conceptual modeling and simulation for analysis: a case example of a highly granular model of an emergency department," in *Simulation Conference (WSC), 2013 Winter*. IEEE, 2013, pp. 2398–2409.

[21] P. Battu Hanumantha Rao, K.Padmaja, "A brief view of model based systems engineering methodologies," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 4, no. 8, 2013.

[22] J. O. Grady, "Universal architecture description framework," *Systems Engineering*, vol. 12, no. 2, pp. 91–116, 2009.

[23] A. C. S. Mellor and T. Futagami, "Model driven development," *IEEE*, pp. 14–18, 2003.

[24] A. P. Sage and W. B. Rouse, "Handbook of systems engineering and management," *John Wiley and Sons*, pp. 1–2, 2009.

[25] D. R. Cox and E. J. Snell, "Applied statistics-principles and examples," *CRC Press*, vol. 2, 1981.

[26] B. D. Ripley, "Pseudo-random numbers," *Wiley Online Library*, pp. 14–52, 2008.

[27] K. Binder, S. Krauss, and G. Bruckmaier, "Effects of visualizing statistical information–an empirical study on tree diagrams and 2× 2 tables," *Frontiers in Psychology*, vol. 6, p. 1186, 2015.

[28] J. D. Andrews and S. J. Dunnett, "Event-tree analysis using binary decision diagrams," *IEEE Transactions on Reliability*, vol. 49, no. 2, pp. 230–238, 2000.

[29] J. Lauks, B. Mramor, K. Baumgartl, H. Maier, C. H. Nickel, and R. Bingisser, "Medical team evaluation: effect on emergency department waiting time and length of stay," *PloS One*, vol. 11, no. 4, p. e0154372, 2016.

[30] J. Crane, C. Noon *et al.*, "The definitive guide to emergency department operational improvement: employing lean principles with current ed best practices to create the no wait department," *CRC Press*, pp. 1–2, 2011.