HospiT'Win : designing a discrete event simulation-based digital twin for real-time monitoring and near-future prediction of patient pathways in the hospital

Abdallah Karakra

THÈSE

en vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

délivré par
IMT – École Nationale Supérieure des Mines d'Albi-Carmaux

---

présentée et soutenue par

Abdallah Karakra

le 18 October 2021

HospiT'Win: Designing a Discrete Event Simulation-Based Digital Twin for Real-Time Monitoring and Near-Future Prediction of Patient Pathways in the Hospital

---

École doctorale et discipline ou spécialité :
EDSYS : Génie industriel et informatique

Unité de recherche :
Centre Génie Industriel, IMT Mines Albi

Directeurs de thèse :
Jacques Lamothe, Professeur, IMT Mines Albi, (Directeur)
Franck Fontanili, Maître-assistant, IMT Mines Albi, (Co-directeur)

Autres membres du jury :
Virginie Goepp, Maître de Conférences HDR, INSA de Strasbourg, *(Rapporteure)*
Gregory Zacharewicz, Professeur, IMT Mines Alès, *(Rapporteur)*
Thibaud Monteiro , Professeur, INSA Lyon, *(Président)*
Sondes Chaabane, Maître de Conférences , INSA Hauts-de-France, *(Examinatrice)*
Adel Taweel, Professeur, Birzeit University et King's College London, *(Examinateur)*
Elyes Lamine, Maître de Conférences HDR, IMT Mines Albi et INU Champollion, *(Encadrant)*
Clair Augsburger, Gérant de l'entreprise FlexSim Solutions Logicielles, France, *(Invité)*

Abstract

Hospitals are demanding workplaces that have real-time services and require extensive human interaction at the resource level (doctors, nurses, etc.), the location level (exam rooms, operating rooms, etc.), the process level (pathways), and the user level (patients). Designing and managing such a health care facility can be inherently challenging due to the critical nature of the services and their wide variety and variability, as well as the difficult adjustment to a partially unforeseeable demand of care.

To increase the efficiency and the quality of care while curbing healthcare costs, a hospital requires decision-making support tools. These can be based on organizational engineering methods and tools for monitoring the current state of the organization in real time, for predicting its behavior in the near future, and for improving its processes. Among the tools, discrete event simulation (DES) is able to model the operational process behavior and to assess performance by simulating a chain of events that occur over time. However, despite important features provided by DES software tools, they are often limited to building "offline" simulation models that are not connected to the real world in real time. These simulation models may not be suitable for retrieving the current state of the organization, and they cannot be considered a "Digital Twin". Furthermore, these simulations start with an "empty" and "idle" state, which can be different from the real-world state, and imply a bias in the statistics reports at the end of the simulation run.

This research work deals with a DES-based Digital Twin (DT) approach. It is based on DES models which are used (1) for real-time and online monitoring of patient pathways, and (2) for near-future offline prediction when facing unexpected behavior or unpredictable situations. The major goal of this research is to provide a framework for building a Digital Twin of patient pathways that health care practitioners and decision makers can use as a decision support tool. Some specific issues are also addressed: initialization of the DES models, real-time synchronization with the real world, and the connection between monitoring and prediction models. As a proof of concept, experiments are carried out using an emulator of a hospital service that is connected to a Digital Twin that follows our approach.

Resumé

L'hôpital est un lieu de travail exigeant qui nécessite une prise en charge en temps réel du patient et une interaction humaine élevée au niveau des ressources (médecin, infirmières, etc.), de l'emplacement (salle d'examen, salle d'opération, etc.), des processus (parcours de soins et de santé) et au niveau des usagers (patients). Concevoir et manager un établissement de santé peut être difficile. D'une part, parce que les services peuvent être des ressources critiques, et présenter une grande variété et une grande variabilité. D'autre part, l'adaptation à une demande de soins partiellement imprévisible peut se révéler très complexe. Pour augmenter l'efficacité et la qualité des soins tout en limitant les coûts, un hôpital a donc besoin d'outils d'aide à la décision. Ils peuvent être basés sur des méthodes et des outils d'ingénierie organisationnelle afin de suivre l'état des services en temps réel, de prédire leur comportement dans un futur proche, et d'améliorer les processus. Parmi les méthodes et outils disponibles, la simulation à événements discrets (SED) permet de mieux comprendre le comportement des processus opérationnels et d'évaluer leur performance en simulant une chaîne d'événements qui se produisent dans le temps. Cependant, bien que les logiciels de SED disposent de multiples fonctionnalités, ils se limitent principalement à la construction de modèles de simulation « hors ligne » qui ne sont donc pas connectés au monde réel en temps réel. Par conséquent, les modèles ne sont pas adaptés pour récupérer l'état courant d'une organisation à un instant précis, ils ne peuvent donc pas être considérés comme des doubles numériques (DN). De plus, les simulations démarrent dans un état « vide » et « inactif », qui peut être différent de l'état réel, ce qui peut entrainer un biais dans les rapports statistiques à la fin de la simulation. Notre travail de recherche propose une approche de double numérique à base d'un simulateur à événement discret. Notre double numérique fournit une représentation virtuelle et en temps réel qui est synchronisée avec les ressources physiques et/ou les activités des processus. Il est basé sur des modèles de SED qui sont utilisés pour (1) le suivi en temps réel et en ligne des parcours des patients, et (2) la prédiction hors ligne du futur proche face à un comportement inattendu ou à des situations imprévisibles. L'objectif principal de cette thèse est de fournir un cadre pour la construction d'un double numérique du parcours des patients que des professionnels de santé et des décideurs pourraient utiliser comme outil d'aide à la décision. Plusieurs problématiques spécifiques sont également abordées : l'initialisation des modèles de SED sur l'état courant, la synchronisation en temps réel avec le monde réel, et la connexion entre le modèle de suivi et le modèle de prédiction. Comme preuve de concept, nous proposons des expérimentations basées sur un émulateur d'un service hospitalier connecté à un double numérique développé suivant notre approche.

This thesis was co-funded
by the Occitanie Region
and IMT Mines Albi.
Thanks for their support.

Acknowledgements

It is with great pleasure that I reserve this page to convey my profound gratitude to all of the supportive people in my life who have given me the strength, encouragement and support that has enabled me to survive the difficult times during my PhD journey. This thesis has become a reality thanks to their generous help and support.

First and foremost, I am extremely grateful to my supervisor Dr. Franck Fontanili for his invaluable advice and continuous support. He was very patient, motivated and always provided me with excellent guidance. I cannot have imagined having a better advisor and mentor for my PhD studies.

My sincere thanks also go to Dr. Elyes Lamine for the time he spent sharing his wealth of knowledge with me. He provided me with an abundance of scientific advice and knowledge as well as many insightful discussions and suggestions.

I would like to convey my sincere thanks to Prof. Jacques Lamothe for his valuable guidance, support, and his great efforts to coordinate this research work, as well as the time spent attending all of my meetings.

I would like to thank all the other members of the jury for accepting to evaluate my work. Special appreciation goes to the committee president, Prof. Thibaud Monteiro, for accepting to chair my PhD defense, to review committee members Dr. Virginie Goepp and Prof. Gregory Zacharewicz for their interest and to Prof. Adel Taweel and Dr. Sondes Chaabane for being my thesis examiners. I would also like to thank Clair Augsburger, Allister Wilson, and Markus Cueva from FlexSim company, as well as Bob Mcllvride from Skkynet company and Sebastien Rebiere, and Ludovic Pouzenc from IMT MINES ALBI for their technical support of my study. Great thanks go to Paul Gaborit for his valuable help in using LaTeX to write this PhD manuscript, as well Great thanks go to Shauna Brown for her efforts reading and editing this thesis manuscript.

For the kind secretaries Claude Laffore and Marlène Boval: Thank you very much for your strong support and nice work during this PhD. Heartfelt thanks go directly to the dear Isabelle Fournier: you were the first one to start working on the documentation to bring me here. Unfortunately, I did not have the good fortune to meet you in person due to your illness, then you passed away. However, you remain in my heart. A great thanks go to Marie Christine PLANTIE, Severine Pefaure, and Rafika Thabet for their support in many administrative stuff.

I also want to give special thanks to all of those in the computer science department of Birzeit University, Palestine, for their encouragement and support for me to continue this PhD degree. Thank you, Dr. Yousef Hassouneh, Mr. Nael Qaraeen, Mr. Hafez Barghouthi, Dr. Ahmed Abusnaina, Dr. Majdi Mafarja, Dr. Sobhi Ahmad, Mr. Wahbeh Mousa, Dr.

Mamoun Nawahdah, Dr. Samer Zein, Prof. Mustafa Jarrar, Dr. Bassem Sayrafi, and others who I may forgetten due to the writing-a-thesis-distortion phenomenon; thank you all. Special thanks to Dr. Ahmad Alsadeh for accepting to be my supervisor for my master's thesis, which was the first step toward my PhD.

My warmest thanks go to my friends and colleagues at IMT MINES ALBI, for their support in my academic research and my daily life. Thanks to Ibrahim, Asaad, Mustapha, Khaled, Nafee, Sina, Sanaa, Rania, Raphaël, Audrey, Mahsa, Guillaume, Mehran, Julien, Eva, Aurélie, Quentin, Antoine, Sam, Robin, Romain, Oussema, Manon, Delphine, Liwen, Thibaud, and others who I may forgetten due to the above-mentioned phenomenon. Special thanks go to Leah, Marine, and Clara, for helping me in French translation for this PhD manuscript. Again, thank you all. I'd like to thank everyone in the CGI lab started by Prof. Xavier Lorca.

I am deeply and forever indebted to my parents, my wife, and my children for their love, support and encouragement throughout my entire life. Without their tremendous understanding and encouragement in the past few years, it would have been impossible for me to complete my studies. I am also very grateful to my brothers and sisters. I dedicate this thesis to all of you. Last but not least, I am grateful to other friends who have been there for me through all of the ups and downs.

. . . Thank you all once again!

To my parents
To my lovely family, my wife and my children
To my brothers and sisters
For their unconditional support and love.
I love you all.

Summary

Abbreviations

DT	**D**igital **T**win
CPS	**C**yber- **P**hysical **S**ystem
DES	**D**iscrete **E**vent **S**imulation
DTM	**D**igital **T**win for **M**onitoring
DTP	**D**igital **T**win for **P**redicting
DTO	**D**igital **T**win for **O**ptimization
LOS	**L**ength **O**f **S**tay
IoT	**I**nternet **o**f **T**hings
AI	**A**rtificial **I**ntelligence
ML	**M**achine **L**earning
RTLS	**R**eal **T**ime **L**ocation **S**ystem
RFID	**R**adio **F**requency **I**dentification
RTD	**R**eal **T**ime **D**ata
KPI	**K**ey **P**erformance **I**ndicator
VA	**V**alue **A**dded
N-VA	**N**on-**V**alue **A**dded
ENVA	**E**ssential **N**on-**V**alue **A**dded
SCA	**S**tate **C**ollection **A**pproach
BSA	**B**ase **S**imulation **A**pproach
WL	**W**aiting **L**ine
RD	**R**egistration **D**esk
WR	**W**aiting **R**oom
ER#	**E**xam **R**oom #

1 Introduction

The domain of health care is one of the largest in the world (Bhat et al., 2014). As claimed by (World Bank, 2021), global health care spending was roughly 10% of total gross domestic product (GDP) in 2018. The United States invested the most per capita, totaling 10,623.85 US dollars, followed by Switzerland. Canada is one of the top 20 countries in the world investing in health per capita, with a total of 4,994.90 US dollars, whereas the global average was 1,111.082 (World Bank, 2021).

Nowadays, health systems are under pressure on all continents, in both developed and developing countries. They are unable to address the medical needs of the population under their care, which are growing in conjunction with increased life expectancy and the prevalence of chronic diseases. Furthermore, governments are constantly defining new directives for more rigorous management of their health care systems in terms of cost efficiency and care quality. Hence, the hospitals at the heart of these health systems are facing difficulty in responding quickly to the continuously changing health care needs of a healthcare environment that is becoming more and more demanding and unpredictable. For instance, resource planning in a complex system such as a hospital is increasingly challenged by the diversity and the variability of care services and increasing demand, as well as the constantly changing availability of resources.

Currently, unexpected events are regularly seen in healthcare organizations, especially in hospitals. Patient no-shows to scheduled appointments, patient or doctor delays, surgical intervention delays or postponements due to emergencies, a lack of nurses, or a rapid incoming flow of patients due to a bus accident are some examples of unexpected events in a hospital. As a consequence of these events and the inadequacy of resources for meeting demand, many patients end up waiting at the registration desk, services or procedures are duplicated, waiting times are prolonged in the waiting rooms, staff are overloaded, and so on. Some of the previous events could be classified as internal events because the main cause of these events came from inside the organization. The other type of event could be classified under the category of external events because the main causes of these events came from outside the healthcare organization. For example, an accident that happens on the highway due to weather conditions, when at the same time there is a lack of beds in the emergency department. All of these events will affect hospital functionality, patient care, patient satisfaction, and the quality of health care in general.

At this stage, it is worth noticing that the quality and the relevance of the decisions that should be made regarding these unexpected events are strongly linked to the visibility that decision-makers have on the given situation and its future development. Hence, the real-time monitoring of patient pathways, which are a sequence of clinical, non-clinical, and administrative activities, is one of the most important aspects in delivering crucial information to the hospital staff and managers. This information helps to determine and assess the efficient medical and administrative services that can be used to provide the right care for each individual patient at the right time and in the right location (Huang et al., 2016). Monitoring patient pathways could provide information related to the duration the patient spends in each activity, the distance covered by the patient to arrive at a specific activity (Araghi et al., 2018), the number of patients at each activity, the quantity of human resources needed to execute the activity, the type and the state of fixed resources used by this activity, etc. All this information can be used as input to a decision support tool for predicting the near future of the hospital. Based on the outcome results, the hospital manager can make decisions that help improve the quality of healthcare, utilizing resources in an efficient way, managing the increasing number of patients, and increasing patient satisfaction.

In the industrial domain, various systems and software programs have been used for purposes such as monitoring, controlling, planning, etc. For example: Supervisory Control and Data Acquisition (SCADA)(Gaushell et al., 1987) is used for monitoring and remotely controlling machines; Enterprise Resource Planning (ERP)(Moon, 2007) is used for providing managers with a detailed view of the organization's resources; Manufacturing Execution Systems (MES)(Saenz de Ugarte et al., 2009) are used for tracking the transformation of raw materials into finished goods; Enterprise Asset Management (EAM)(O'Hanlon, 2005) is used for the maintenance and management of physical assets; and Product Lifecycle Management (PLM)(Stark, 2015) is used for tracking the journey of individual products from design to manufacturing and distribution.

In recent years, considerable literature has appeared on the concept of the Digital Twin (DT). This interest is mainly due to its potential to reduce the cost of system verification and testing, to produce tailored decision support information and alerts for users, and to predict changes in physical systems over time (Madni et al., 2019). Even though the terminology has changed over time, the DT can be loosely described as a simulated **real** environment, based on a strong two-way interaction between the digital and physical worlds and using a set of well-aligned, descriptive and executable models with the aim of supporting decision making during design and operation stages (Roland Rosen, 2018).

The objectives of a DT applied to patient pathways are very ambitious. Scientific research in the healthcare field, compared to what has been published in the manufacturing field, is scarce and is still in an early developmental stage (Barricelli et al., 2019). There is no methodological framework to structure and guide the design of a DT according to hospital decision-maker needs (Saracco, 2019). Currently, in the industrial domain, DT takes advantage of the Internet of Things (IoT) and Artificial Intelligence (AI) solutions to perform in-depth analyses. This is very helpful in finding potential challenges, minimizing downtime, reducing maintenance costs, simulating future scenarios and customizing production based on customer needs, etc.

Our research sets out to investigate the usefulness of DT for hospital management and its practical considerations. It contributes to setting the first foundations of a new framework called HospiT'Win, and ascertains the process of building a DT for monitoring and predicting patient pathways in real time. This process includes a way of initializing and synchronizing the DT with a real-world state and events, and then predicting the near future.

1.1 Business Questions

There are three key features of the DT which can be used in health care in general and for patient pathways in the hospital in particular. This section illustrates several business questions to demonstrate the idea behind these features.

1. **Monitoring events in real time:** having a real-time view of the hospital's current state will help the decision-maker and/or hospital manager to recognize and understand the complexities of pathways and service dependencies. Based on this, they can manage patient pathways in an efficient manner. Indeed, real-time monitoring for the pathways will help answer many questions that could be relevant for managing pathways, for example:
   (a) How many patients are on the hospital premises right now?
   (b) What is the current location of each patient?
   (c) How much time has each patient spent in the waiting room?
   (d) How much time has each patient spent with the doctor?
   (e) What is the length of stay of each patient in the hospital (LOS)?
2. **Predicting the outcome of the near future and evaluating a specific scenario:** this enables the hospital manager and the decision-maker to respond to up-to-date situations or to react to current events. Having a vision of what will happen in the future is the best way to improve the current situation. Furthermore, exploring a range of future scenarios might assist in determining how robust the decision-maker's and/or hospital manager's strategies are in the face of uncertainty. This type of analysis may also reveal which tactics will be the most adaptable or resilient in the face of future change. Moreover, predicting the future will aid in addressing a variety of questions, such as how to respond quickly to certain events as they occur. The following are some of these types of questions:
   (a) What will happen if the hospital has a very rapid incoming flow of patients into the emergency department?
   (b) How will the increasing number of patients affect hospital services, staff, assets, utilization, and more?
   (c) What will happen if the patient does not respect his/her appointment?
   (d) What will happen if the doctor does not arrive on time (delay)?
   (e) What will happen if the incoming patients exceed the capacity of the waiting rooms?
   (f) How can the impact of an unexpected event in the operating rooms (emergencies or delays) be evaluated?
3. **Improving the quality of health care:** the idea is to gather knowledge on the appropriate health care for specific situations and to adapt it to similar situations. Answering different questions such as the following could improve the quality and the outcomes of the health care:
   (a) How can hospital resources be managed in an efficient manner?
   (b) How can operational costs be minimized?
   (c) How can waiting times and/or delays be reduced?
   (d) How can patient satisfaction be maximized?

To achieve these features, we believe that a Digital Twin-based decision support tool to aid the hospital manager to make real-time decisions according to current situations and based on visions of the future is needed. Indeed, the DT technique seems to be a relevant means for meeting these needs. As will be detailed further in this thesis, it consists of virtual representations of real assets and/or processes that are used to understand, predict and optimize operations and efficiency.

1.2 Research Problems

There is a wide range of studies in the literature that demonstrates the idea and the interest of simulation modeling. Numerous studies have been conducted over the last twenty years. In the healthcare area, these studies present different types of simulation techniques and applications (Salleh et al., 2017). Discrete Event Simulation (DES) is one of the many tools and methods that are used in the analysis and the improvement of healthcare services (Kammoun et al., 2014) (Zhang, 2018). Indeed, DES provides perhaps the most powerful and intuitive method for analyzing, evaluating, and improving complex healthcare systems (Jacobson et al., 2006). There are many application domains where DES has been used. For example, in hospital supply chains (Kammoun et al., 2014), outpatient clinics (Al-Araidah et al., 2012), emergency departments (Connelly et al., 2004), intensive care units (Z. Zhu et al., 2012), and more.

The problem is that the traditional DES is heavily dependent on historical data (Hammad et al., 2011; Tavakoli et al., 2008), which means the simulation models use statistical data and probability distributions to estimate task duration. This leads to inaccurate and less reliable prediction. In other words, because the data can be extracted from a non-significant period in the past, running the traditional DES model, which is called an **offline** DES model, to simulate the near future could give the wrong results for predictive analysis and decision making. This is a critical issue in dynamic and complex systems like health care in general and the hospital in particular. Moreover, these simulations start with an **empty** and **idle** state (Hanisch et al., 2005), which can be different from the real-world state. This may imply a bias in the statistics reports at the end of the simulation run, mainly if the run is short (a few minutes or hours). Furthermore, gathering, processing and preparing the data to be used in the simulation model is time consuming, even if the data is available in the information system. In addition to these drawbacks, working with offline DES requires experience, expense, and time for adjusting and calibrating the model.

In the hospital, many parameters and factors are continuously changing, such as: the number of patients, the number of demands, staff scheduling, activity duration from one patient to another, etc. This leads to the fact that the traditional DES model may not deal with all of these changes in real time, and the prediction outcome will not be reliable for short-term assessment.

This thesis benefits from the concept of a DT based on DES and real-time data to answer the following research questions:

1. **How can DES be used to design a decision support tool based on a Digital Twin for real-time monitoring of patient pathways in hospitals and predicting their near future?**
   (a) What models should be designed in building a Digital Twin?
   (b) How should such models be validated?
2. **How can the Digital Twin have an accurate vision of the current situation of the patient pathways?**

(a) How can the DT be initialized with current information about the state of real patient pathways?

(b) How can the virtual patient pathways based on discrete events be continuously synchronized to have the same state as the real patient pathways?

3. **How can the Digital Twin for Monitoring (DTM) be triggered to run the Digital Twin for Predicting (DTP)?**

(a) In which cases/situations does the predictive model work (reactive, proactive or on demand)?

1.3 Contribution of This Thesis

Because the traditional DES model is dependent on historical data, and as these data represent certain and particular situations in the past and may not deal with current changes, it can be difficult to use this model to predict and anticipate future events accurately, mainly in the very short term. This thesis sets out to investigate and to prove the concept of the usefulness of the DT technique for hospital management. It points out the process of developing a DT framework based on **online** DES for running the DES model in parallel with the real world in real time. This framework is dedicated to **real-time monitoring of patient pathways and predicting their near future**. It aims to handle irregular, unusual and unexpected behaviors that may occur in hospitals and it helps to make the right decision to mitigate unpredictable situations. Different issues related to the way of developing, initializing and synchronizing the DT are discussed. We summarize our contributions in Table 1.1. The first two contributions are scientific, while the third contribution is technical. The third contribution allowing us to practically implement and demonstrate the first two contributions.

Contribution	Description	Achievement	
Methodology for developing a DT for patient pathways inside the hospital.	This methodology aims to help in designing a discrete event simulation-based DT for real-time monitoring and near-future prediction of patient pathways in the hospital.	I. Life cycle to design the DT of patient pathways. II. Meta-model for real patient pathways in hospital. III. DT meta-model for patient pathways in hospital. IV. Process flow modeling notations.	Chapter 3
Mechanisms for initializing and synchronizing the DT of patient pathways with the real patient pathways.	The aim of these mechanisms are: I. To initialize the DT models (DTM and DTP) with the current state of real patient pathways. II. To synchronize the DTM to run in parallel with real patient pathways.	I. Algorithm for initializing the DTM and the DTP. II. Algorithms for synchronizing the DTM with the real patient pathways. III. GRAFNet chart to illustrate the different concepts behind synchronization.	Chapter 4
A proof of concept for the monitoring and predicting DT models.	This contribution aims to adopt and assess the suggested methodology as well as the developed initialization and synchronization algorithms to monitor the patient pathways and anticipate their near future.	I. Experimental platform. II. Dashboard. III. Digital twin for monitoring. IV. Digital twin for predicting.	Chapter 5

Table 1.1: Thesis contribution

1.4 Organization of This Thesis

This thesis is structured in the following way:

Chapter 2: Describes the overall topics that are related to the core of our research, such as patient pathways, DES, and the DT. However, different investigated topics are demonstrated, such as the main components, the main usage, the application domains, and the characteristics of the aforementioned topics. In addition, this chapter discusses the approaches used for initializing DT models and synchronizing them with the real world. The limitations and the characteristics of these approaches are presented.

Chapter 3: The purpose of this chapter is to introduce and discuss the research methodology for designing and building the DT of patient pathways. The theoretical and technological concepts behind building the DT have been illustrated. Moreover, different techniques have been developed here, such as: a meta-model for real patient pathways in the hospital, a meta-model of a DT for patient pathways in the hospital, and a rich process flow that helps in designing and developing a DT for patient pathways of the hospital.

Chapter 4: This chapter discusses the various concepts behind the initialization and the synchronization of the DT models of patient pathways with the real patient pathways. Some algorithms and extensions of graphical modeling languages, such as Petri Nets and GRAFCET, have been merged and used to clarify and demonstrate these concepts. In addition, different parameters to be taken into account when initializing the DTM and the DTP are highlighted. Moreover, various issues related to the initialization and the synchronization have been solved.

Chapter 5: The aim of this chapter is to follow the proposed methodology in Chapter 3 and use the initialization and synchronization algorithms that were discussed in Chapter 4 to design a proof of concept for the DTM and a prototype for the DTP. To achieve this goal, the developed experimental platform has been used.

Chapter 6: This chapter summarizes the main objective of this research work, along with the research questions and their answers. In addition, this chapter demonstrates research challenges, limitations, future work, and recommendations. Due to the importance of securing the DT, some significant points related to cybersecurity have been illustrated in the future work of this chapter.

Figure 1.1 illustrates the core chapters of this thesis that make up the proposed approach, as well as the answers to the research questions.

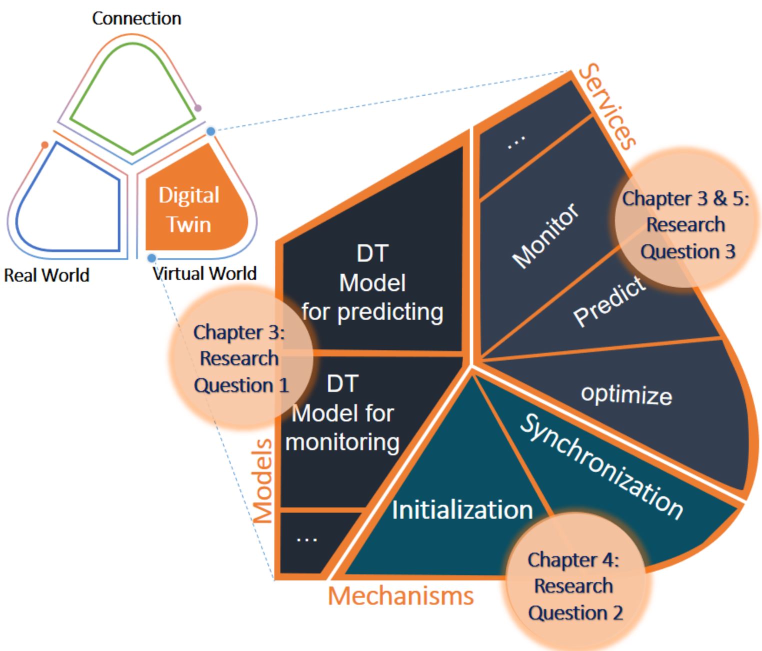


Figure 1.1: Core chapters of the thesis

2

Background and Related Works

2.1 Introduction

In Chapter 1, we have discussed the main objective of this research work, which is summarized by using online Discrete Event Simulation (DES) to design a proof of concept for a Digital Twin (DT) of the patient pathways in the hospital, named HospiT'Win. This DT is able to monitor the patient pathways in real time and predict their near future.

To achieve the objective of this research work, a literature review related to our work has been done. Understanding the patient pathway in a hospital is an essential step, as it is considered to be one of the main concepts of hospital management. In fact, understanding the main elements of patient pathways, the relations between these elements and the tools used to improve these pathways is the first step towards our goal. Understanding the related work on DES is the second step on the road for achieving our objective. Indeed, our DT is based on DES, so understanding the main components of DES, the relations between these components and the way of utilizing these components in designing and developing a realistic, high-fidelity, real-time, and dynamic simulation model is another important step just before the last step in the direction of our goal.

Last but not least, the DT is an old-new concept, and it is the core of this research work. So a comprehensive review of this concept is mandatory for our research. For example, investigating the literature to fully understand the main layers behind this concept, the technology used to build it and the way of using it will help us design our type of DT.

The purpose of this chapter is to share knowledge from the past and investigate the current state of research on DTs and their applications in the healthcare domain. This review will concentrate on three main concepts that appear repeatedly throughout the reviewed literature: patient pathways in hospitals, DES, and DTs. Various technologies and approaches that are used with these concepts will be depicted.

2.2 Patient Pathways

Nowadays, decisions concerning healthcare operations are influenced by several objectives (Sarno et al., 2016): 1) improving the quality of care, 2) minimizing operation cost, 3) maximizing usage of resources, and 4) managing the increasing number of patients. Therefore, the efficient use of healthcare resources is a fundamental point, particularly during and after global economic crises (Stefanini et al., 2016).

The patient pathway is one of the effective management tools that have been used to improve the quality of care throughout the continuum of improving risk-adjusted patient outcomes, promoting patient safety, increasing patient satisfaction, and maximizing resource utilization (Schrijvers et al., 2012). In fact, the patient pathway could be used as a consistent treatment schedule or a framework for managing medical activities in order to maximize the utilization of medical resources while increasing service quality. It has several advantages, including improved medical care continuity, increased practice consistency, and well-monitored care protocols, all of which contribute to increasing the quality of medical treatment (Srivastava et al., 2020).

The aim of this section is to provide answers to the following questions: (1) What are patient pathways in a hospital? (2) How might the patient pathways in hospitals be used to improve healthcare quality? (3) What are the various methods for improving patient pathways in hospitals? (4) Why is it necessary to develop new technology in order to improve patient pathways?

Most early studies as well as current works have shown that a common definition of patient pathways is not currently available (Richter et al., 2019). Some previous studies used various pathway terms as synonyms for patient pathways. Mainly, these are treatment pathways (Basketter et al., 2018; Manning et al., 2017; Slaich et al., 2018), care pathways (Basketter et al., 2018; Bowers, 2009; Corless et al., 2018; Despiau et al., 2015; Porritt et al., 2012), and patient journeys (Swancutt et al., 2017; Thompson et al., 2015; Tritter et al., 2014). Similarly, (Dey et al., 2013; Vanhaecht et al., 2010) pointed out that the critical care pathway, the integrated care plan and the care map are identical.

(Richter et al., 2019) did a systematic review to understand the patient pathways in the context of integrated healthcare. They argue that patient pathways differ from other pathway terms. Moreover, many studies were related to pathways in healthcare. As a result of these studies, different terms for pathways have appeared, such as patient journey (Gualandi et al., 2019), clinical pathway (Kinsman et al., 2010), patient flow (Tlapa et al., 2020), care pathway (Pinder et al., 2005; Schrijvers et al., 2012), critical care pathway (Wigfield et al., 1996), integrated care plan (Hägglund et al., 2005) and care map (Dickinson et al., 2000). Thus, according to (Halvorsrud et al., 2019), few works in the literature have used patient pathway, patient journey, clinical pathway, patient flow, or care pathway in a similar manner.

To the best of our knowledge, no general agreement has been achieved on what a patient pathway is and on what the other healthcare pathways do and what they do not do. The pathway terms in health care remain unclear due to the different definitions, opinions, and characteristics that exist in prior studies. Table 2.3 shows the common definitions of patient pathways.

A study done by (De Bleser et al., 2006) develops a general definition of the pathways in healthcare. This study found that the most widely used and internationally accepted term is clinical pathways. As a conclusion to this study, **a clinical pathway** was defined as "a method for the patient care management of a well-defined group of patients during a well-defined period of time. A clinical pathway explicitly states the goals and key elements of care based on EBM (Evidence Based Medicine) guidelines, best practice and patient expectations by facilitating the communication, coordinating roles and sequencing the activities of the multidisciplinary care team, patients and their relatives by documenting, monitoring and evaluating variances and by providing the necessary resources and outcomes."

Based on the aforementioned definition and the definitions in Table 2.3, we have selected the most common keywords (terms) based on word frequency. As result, the network graph represents the terms with high-use frequencies and their relationships, as illustrated in Figure 2.1.

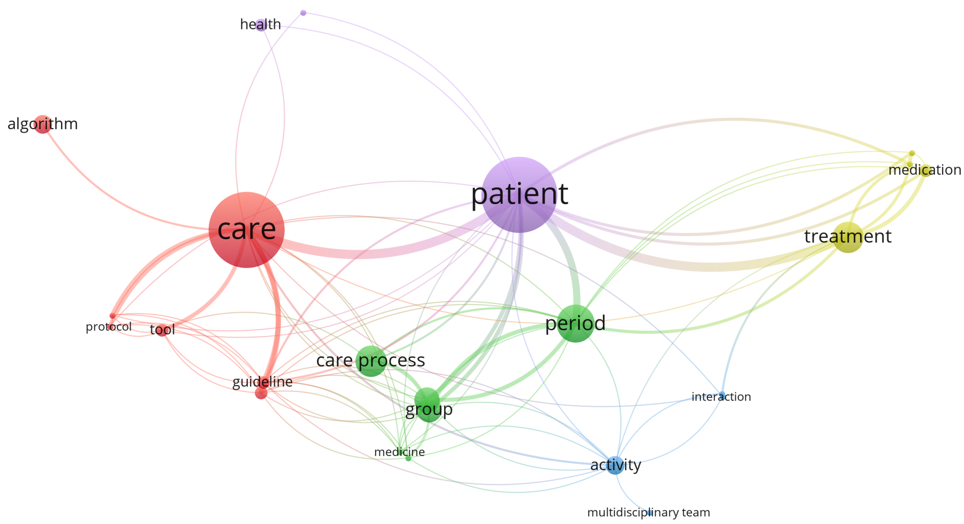


Figure 2.1: Network graph for the high-frequency terms of patient pathway definitions

This graph indicates the high-frequency terms that exist in the definition of the patient pathways that appear in proximity. To have a closer look at which keywords are critical in these definitions, a density visualization of the words has been developed, as shown in Figure 2.2.

Definition	Term	Source
“A comprehensive, coherent description of one or more patients’ contacts with different parts of the health-care system during a period of illness.”	Patient journey, patient pathway, clinical pathway, patient flow, and care pathway	Defined by NDH (Halvorsrud et al., 2019)
“The chronological chain of events that constitutes the patient’s encounter with various parts of health and care services.”	Critical pathways, care paths, integrated care pathways, case management plans, clinical care pathways, or care maps	Defined by NMHCS (Halvorsrud et al., 2019)
“A task-oriented description of a clinical procedure that outlines the essential steps and decisions to be taken for offering optimal care to patients presenting a specific clinical condition.”	Integrated care pathways, coordinated care pathways, care maps, or anticipated recovery pathway	(Vanhaecht, 2007)
“A complex intervention for the mutual decision making and organization of care processes for a well-defined group of patients during a well-defined period.”	Care pathway	(Schrijvers et al., 2012)
“Task orientated care plans which detail essential steps in the care of patients with a specific clinical problem and describe the patient’s expected clinical course.”	Clinical pathway	(Hudson, 2011)
“A standardized algorithm of a consensus of the best way to manage a particular condition. Modalities used: teletherapy, brachytherapy, hyperthermia and stereotactic radiation.”	Clinical pathway, critical pathway, treatment pathway, clinical medicine	(dictionary, 2011)
“The interaction among activities (treatments and cares), resources (medical staff, medical equipments, operating rooms, etc.), and requirements from the stakeholders.”	Care plan, clinical pathways	(Bernardi et al., 2019)
“A method for the patient-care management of a well-defined group of patients during a well-defined period of time.”	Clinical pathway	(De Bleser et al., 2006)
“Schedules of medical and nursing procedures, including diagnostic tests, medications and consultations designed to effect an efficient, coordinated program of treatment.”	Critical pathways	(Anderson et al., 2004)
“A complex intervention for the mutual decision making and organization of care processes for a well-defined group of patients during a well-defined period.”	Care pathway	Defined by EPA (Larsen, 2012)
“A mathematical system model that is being designed and manipulated to support reengineering an existing real-world clinical process.”	Clinical pathway	(B. Zeigler et al., 2019)
“A structured, multidisciplinary patient care plan in which diagnostic and therapeutic interventions performed by physicians, nurses, and other staff for a particular diagnosis or procedure are sequenced on a timeline.”	Clinical pathway	(Ireson, 1997)

Note: +The term (EPA) stands for European Pathway Association.
Note: +The term (NDH) stands for Norwegian Directorate of Health.
Note: +The term (NMHCS) Norwegian Ministry of Health and Care Services

Table 2.1: Definitions of patient pathways

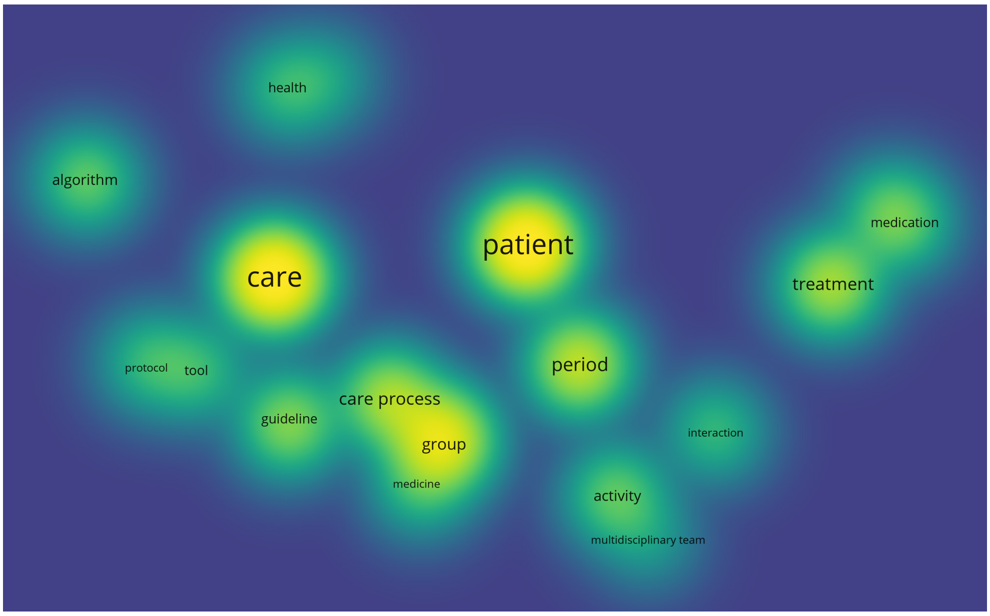


Figure 2.2: Keyword density visualization of words in the patient pathway definitions

Based on this graph, there are different views that could be seen for patient pathways:

- By looking at the terms of **care, medication, treatment, medicine and health**, it seems that patient pathways represent only the clinical activities (e.g., diagnostics, treatment purposes, etc.) in a healthcare organization.
- From the term **group**, which is related to "a well-defined group of patients in a well-defined period", different types of pathways are dependent on the group types. In this case, each group of patients could follow a specific pathway based on the type of illness.
- Another view could be based on the terms **algorithm, protocol, guideline and care process**. In this view, the patient pathway can represent protocols/algorithms that define the sequence of the clinical activities that the patient has to follow (for example, after diagnosis, the patient must go to the waiting room, and so on.)
- Based on the term **multidisciplinary team** that refers to a "collective involving health care providers from more than one discipline" (Dictionary, 2011), the patient pathway could be illustrated as a high-level view that includes all of the health care organizations that the patient may visit.

Considering this analysis of the keywords, we observe that none of the previous definitions totally fit the scope of this research work. In this setting, we therefore propose the following definition :

> **Proposed definition for patient pathways in the hospital**
>
> A patient pathway is a sequence of activities required for caring for a patient, undertaken in the hospital, beginning with admission and ending with discharge. These activities might be clinical (diagnostic, for example), non-clinical (waiting or moving, for example), or administrative in nature (registration, etc.).

2.2.1 The effectiveness of patient pathways in the hospital

The patient pathway is considered one of the effective management tools that have been used by healthcare professionals to identify the best processes and the best utilization of healthcare resources in their healthcare organizations (Guus Schrijvers, 2012; Panella et al., 2003). (Kitchiner et al., 1996) point out that "Pathways can specify and prospectively monitor important aspects of care in up to 80 % of patients". For example, monitoring the patients in the pathways allows the complexities and service dependencies inside the health organizations to be discovered (Karakra et al., 2018).

From an organizational standpoint, one of the key roles for achieving the required levels of service and efficiency is to manage the resources of the organization (Stefanini et al., 2016). Therefore, continuous monitoring of patients in their pathways provides an in-depth knowledge of the flow of the patients, the state of the patients, the location of the patients, the activities visited by the patients, and the amount of time the patients spend at each activity (De Bleser et al., 2006). Patient pathways are also associated with indicators that healthcare professionals can use to improve and manage the quality of health care in terms of managing resources, scheduling staff, patient flow, efficient usage and planning of beds and rooms (Dey et al., 2013). (Kinsman et al., 2010) state that a "pathway is an effort made in order to: outline the steps in detail, outline the important steps that must be taken, describe services to patients and estimate possible clinical problems; it can be used as a system for quality improvement and cost containment". According to (Schrijvers et al., 2012), the patient pathway is considered a tool that can be used to improve the quality of care by promoting patient safety, improving patient satisfaction, and making better use of resources. In the same vein, (Aspland et al., 2019) mention that patient pathways are considered an efficient and effective approach of standardizing treatment progression, supporting patient care, and facilitating clinical decision making.

In short, in order to maximize the usage of medical resources while improving service quality, a patient pathway can be utilized as a consistent framework for managing medical activities. It provides a number of benefits, including improved medical care continuity, increased practice consistency, and well-monitored care procedures, all of which contribute to improving healthcare quality, minimizing operation cost, managing the growing number of patients and increasing patient satisfaction.

In this section, we have answered one of the objective questions, which is: "How might the patient pathways in hospitals be used to improve healthcare quality?" Next, in Section 2.2.2, we are going to answer the question: "What are the various methods for improving patient pathways in the hospital?".

2.2.2 Approaches to improving patient pathways and their limitations

In 1999, the Institute of Medicine (IOM) recommended making healthcare processes more transparent, safe, and organized around patient needs (Donaldson et al., 2000; Leape et al., 2005). Several methods have been suggested for improving the quality, efficacy, and/or safety of health care (Vanhaecht et al., 2007). One of these methods is patient pathways.

Improving the patient pathway is often difficult because it relies on the interaction of numerous diagnoses and diverse departments. Different specialties have their own sets of routines and quality programs that must be followed (Aasebø et al., 2012). Furthermore, the patient pathway during workup faces different levels of delays, such as: patient and doctor delays, waiting time for x-rays and computed tomography (CT) scans, referrals to specialists, and waiting time for radiotherapy or surgery, etc. (Aasebø et al., 2012) have used the **Lean process** as a tool for improving this workup time, which is a method for making the business more effective by eliminating/removing wasteful practices or unnecessary activities. (Gabriel et al., 2019) tried to use **Lean health care** which is an application of Lean ideas in healthcare facilities to increase the number of treated patients and reduce the Length Of Stay (LOS) without compromising the quality of services or patient safety.

Many tools have been used throughout the literature to improve patient pathways. These are classified by (Aspland et al., 2019) into four categories: (1) **stochastic modeling**[1], (2) **data mining** [2] **or machine learning (ML)** [3], (3) **simulation** [4], (4) **optimization** [5]**, and heuristics** [6]. These tools are used alone or combined; for example, using simulation alone or simulation with machine learning. However, their use has been based on offline methods, which means they were not connected to the real world in real time to receive up-to-date data, and their work was based on historical data that was perhaps extracted from a non-significant period in the past, and was possibly not suitable for current situations.

In addition, some process-oriented modeling languages have been used to model patient pathways. From the Management Information System (MIS) viewpoint, patient pathways are typically modeled using common process-oriented modeling languages like flow charts (Braun et al., 2014), Unified Modeling Language (UML), activity diagrams (Spyrou et al., 2005), Discrete Event System Specification (DEVS) (B. Zeigler et al., 2019), Integrated Definition (IDEF) (Q. Zhu et al., 2018), Petri Nets (Mahulea et al., 2014), Business Process Modeling and Notation (BPMN) (Model, 2011), Event-driven Process Chains (EPC) from ARIS (Nüttgens et al., 1992), or with the help of proprietarily developed languages (e.g., (Burwitz et al., 2013)). As a result, there is no common tool for modeling patient pathways, and it is left to each institution to define its format and level of detail (Halvorsrud et al., 2019).

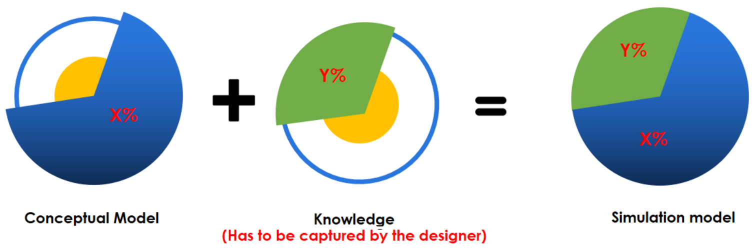


Figure 2.3: Lack of knowledge in the conceptual model

Figure 2.3 illustrates a simple example for creating a simulation model. There are two types of knowledge that are required. One type exists in the conceptual model, but for the second type, the designer of the simulation model must add or find it. Indeed, this knowledge cannot be detailed in this conceptual model, possibly due to the lack of specific modeling notation elements for illustrating this knowledge or for whatever other reason.

[1]Stochastic modeling is a tool for estimating probability distributions of possible outputs by allowing for random variation in one or more inputs over time.

[2]Data mining allows information to be extracted from huge amounts of data. This technique is used to find different patterns inherited in a given set of data in order to generate new, precise and usable data.

[3]Machine learning is a branch of artificial intelligence (AI) that allows computers to learn and improve on their own without having to be explicitly programmed.

[4]A simulation is a time-based representation of the operation of a real-world process or system. Models are required for simulations; the model reflects the key traits or behaviors of the selected system or process, while the simulation depicts the model's progression through time.

[5]Optimization is the process of maximising or minimizing the value of an objective function for some constraint.

[6]Heuristics are any problem-solving or self-discovery strategies that use a practical method that is not guaranteed to be ideal, perfect, or rational, but is sufficient for achieving a short-term objective or approximation.

One of the objectives of this thesis is to identify such required knowledge. This has motivated us to enrich one of the existing works in order to adapt it to our needs in a rich process flow model. In our research work, this model will help in capturing as much knowledge as possible about the patient pathways. This will help the simulation model designer transform the process model into a simulation model of patient pathways, with the objective of reducing the time needed to build such simulation models. However, our simulation model will be different from the traditional simulations, as it will include more knowledge of its connection with the real world. Section 3.4.1.3 provides more details.

In this section, we have answered two questions: "What are the various methods for improving patient pathways in hospital?" and "Why is it necessary to develop new models in order to improve patient pathways?" Next, in Section 2.3, one of the most popular approaches in the literature to improving patient pathways, which is DES, will be discussed.

2.3 Discrete Event Simulation

Because of the importance of the healthcare system, decision-makers seek tools to increase their efficiency and agility. One of these tools is DES. As stated by (Harrell et al., 2000), DES is a reproduction of a dynamic process utilizing a computer model to analyze, measure and enhance the performance of any system without any physical risks or additional costs (BANKS et al., 2010; Montevechi et al., 2007).

The purpose of this section is to provide answers to the following questions:(1) What is DES, and what are its primary components? (2) How can DES be used to improve healthcare quality, and what are the drawbacks? (3) What are the various DES application domains in health care?

2.3.1 Brief description of DES

DES is a type of simulation in which a system is viewed as a discrete collection or sequence of events, where each event occurs at a specific point in time and marks a change in the system's state (S. Liu et al., 2020). Other types of simulation methodologies that have been applied to patient flow management in hospitals are system dynamics, and multi-agent approaches. According to NASA in (Watson, 2018), DES provides a tool that models organizational activities and system responses to discrete events within the operational flow. DES can be used for manufacturing flows, operational processing flows, supply chain flows, and flows of information through an organization (Watson, 2018). The main components involved in DES include the following (Banks, 2005):

- Entities: objects that move, change state and interact with others across the system. Some are permanent because they remain in the system, and others are temporary and stay for a limited period of time.
  - Examples: parts in a factory, customers in a bank, jobs in a processor, patients in a hospital, messages, etc.
- Attributes: common characteristics used to identify entities. The value of an attribute can differ from one entity to another.
  - Examples: arrival time of an entity, color of an entity, priority, etc.
- Variables: reflect the characteristics and the features of the system as a whole. They are not associated with entities, though they can be altered by them. They can be predefined by the simulation software or configured by the user.

  - Example: define a counter to track the current number of entities in the system.

- Resources: a special kind of entity used by other entities to execute and perform an action. For example, an entity requests a resource when allocated and releases it later. Different kinds of resources can exist, such as fixed resources and human resources.

  - Examples: doctors, nurses, desks, computers, x-rays, etc.

- Queues: places where entities wait until they can move forward; for example, because there are no resources available.

  - Examples: Waiting Lines (WL), Waiting Rooms (WR) at the hospital, etc.

- Activities: functions, tasks or actions which resources perform on entities. All activities can have a defined duration of time of a specified length, or an empirical or stochastic distribution, but they can also be defined with a fictitious length.

  - Example: registration activity at the hospital.

- Events: facts that happen at a given time and lead to changes in the state of the process. They can be endogenous, if given by the conditions in the model, or exogenous, if the causes are external.

  - Example: a patient arriving or leaving the hospital, or arriving in a waiting room, or the end of an activity.

- System clock: a global variable used to track simulation time (provides the current value of simulated time).

- Statistics: various probabilistic models are available so that variables, attributes and activity parameters are driven by probabilities. One interest of DES is to assess the impact of uncertainties on simulation behaviour.

- key Performance Indicator (KPI): Typically, the simulation keeps track of the system's KPIs, which quantify the variables of interest.

This section gives only a brief description of the definition of the DES, in addition to its main components. It answers the question "What is DES, and what are its primary components?" The application domains where DES could be applied in health care and its limitations will be discussed in sections 2.3.2 and 2.3.3.

2.3.2 Applications in healthcare systems

Health care organizations have encountered increasing obstacles and problems over the last 40 years. The recent challenges presented by the coronavirus pandemic (COVID-19) are examples of the need to improve the soundness and resilience of the healthcare system in general and hospital management in particular. To achieve this purpose, DES has become a common and efficient decision-making tool for finding an optimum allocation of limited health care resources to enhance patient flow and patient satisfaction while reducing health care costs (Jacobson et al., 2006).

This section discusses numerous works that have been dedicated to the application of DES in the health care sector, and specifically in the hospital: (Cimellaro et al., 2017) developed a DES model that describes and explains the capacity of the hospital emergency department to provide service to all patients following a natural disaster, such as earthquakes or man-made disasters, etc. (DeRienzo et al., 2017) built a DES model to aid clinical managers and administrators in anticipating and scheduling staffing needs in the hospital neonatal intensive care unit using administrative data. (Wong et al., 2016) developed a DES model for a typical 24-hour Accident and Emergency Department (AED) in Hong Kong, and they examined the impact of changes in clinical processes on AED performance using a sensitivity analysis. (Duguay et al., 2007) described a study of using a DES model to reduce the waiting times of patients in the emergency department of Dr. Georges-L.-Dumont Hospital in Moncton (Canada).(Raunak et al., 2009) proposed an architecture for supporting DES that is based on executable process definitions and separate components for specifying resources. In this work, they describe how the proposed architecture might be used to suggest efficiency improvements for hospital Emergency Departments (EDs). (Cubukcuoglu et al., 2020) introduced a DES model for validating a program of requirements for hospital space planning. The main objective of this model is to simulate the procedures of the processing of patients treated by doctors, calculating their throughput and their waiting times based on indicators such as the number of doctors, patient arrivals, and treatment times. Also, this model benefits from hospital design standards to define the space requirements for the patient. (Chan et al., 2019) detailed a simulation study for designing the workflow and improving the process efficiency of the hospital's central kitchen. The authors of this study focused on reducing unnecessary activities and operator movements, minimizing traffic in kitchens, increasing employee efficiency, and analyzing halal and non-halal food segmentation.(Kramer et al., 2020) proposed a design-thinking process that adopts a DES model to make organizational change to the emergency department (ED) located in the north of Italy. To do this, they used a DES model based on historical data.

Table 2.2 lists several works for DES in the healthcare domain. This table shows the area of interest, the aim of the usage, the type of data utilized to achieve the goal and the simulation tool used in each work.

From the numerous works listed in Table 2.2, some conclusions can be drawn. First, the most common uses of DES approaches in hospitals are focused on managing the emergency department and the intensive care unit, which means that the DES model is applied to specific activities in the hospital and little attention is paid to the whole process. For example, there is no focus on patient pathways, which is the core process in the hospital representing the route a patient follows, starting with the arrival at the hospital and ending with discharge. In fact, the pathway includes all the hospital activities the patient will have to visit. From our point of view, monitoring the whole process will help to discover bottlenecks and then remove them.

Second, the DES models used are offline models. These models are run on historical data gathered in the more or less distant past. Indeed, they are not connected to the real world in order to reflect the current situation. Thus, what has been published in the literature does not exactly fit the future. For example, the recent challenges presented by the coronavirus pandemic (COVID-19) are examples of the need to improve DES models. Section 2.3.3 discusses the limitations of DES.

Third, DES is used in the design phase of a system, and is therefore dedicated to long-term decisions: assessing flows, validating requirements and designs, assessing policies and validating organizational changes.

Source	Area of interest	The aim of DES	Type of data	Software used
(Cimellaro et al., 2017)	Emergency department	Evaluate and estimate the resilience of an emergency department in facing disasters.	Historical data (using data collected by a California hospital during the 1994 Northridge Earthquake)	ProModel
(DeRienzo et al., 2017)	Neonatal intensive care unit	Predict and plan for staffing needs in a hospital neonatal intensive care unit using administrative data	Historical data (Duke University Hospital NICU from January 2008 to June 2013)	SAS Simulation Studio
(Wong et al., 2016)	Emergency room	Making decisions on emergency resource planning when there is a sudden surge of emergency patients.	Historical data (June 2009 - June 2010, total number of patient visits: 139,910)	Arena
(Duguay et al., 2007)	Emergency department systems	What-if analysis to reduce patient waiting times and to improve overall service delivery and system throughput	Historical data	Arena
(Raunak et al., 2009)	Emergency department	What-if questions for efficiency improvements of hospital emergency departments (EDs).	Interviews with ED professionals, and analysis of statistical data	JSim
(Cubukcuoglu et al., 2020)	Hospital space planning	What-if scenarios and assumptions on the Program of Requirements (PoR) on space planning can be tested and/or validated.	Not mentioned	Introduced a new plug-in tool for grasshopper algorithmic modeling (GH) in the Rhinoceros CAD program
(Chan et al., 2019)	New hospital central kitchen design	Facilitate short-term and mid-term decision-making in central kitchen resource planning and daily operations	Other hospital kitchen data in addition to some assumptions	FlexSim
(Kramer et al., 2020)	Emergency department	Improving key performance indicators related to patient satisfaction, such as waiting time.	Historical data, on the observation of the current ED situation, and information obtained from ED staff	Not mentioned

Table 2.2: Comparing several DES works dedicated to the healthcare sector

2.3.3 Limitations of DES

The usage of DES in health care has been continuous since the 1960s (Pitt et al., 2008). In fact, there has been a significant increase in its popularity. This increase could be considered clear proof that simulation leads to better decision-making in the management of health care, does not affect patient safety or disturb healthcare activities. As a result, hospitals and health authorities have become increasingly interested in this advantage (Cheng et al., 2016). Despite the presence of important features provided by DES tools, they are limited to building **offline** simulation models that are not connected to the real world in real time. Consequently, the following issues must be addressed:

1. Figure 2.4 illustrates the communication gap between the DES tools and the real world. This means that there is no communication link connecting the DES with the real world in real time. Several features of a DT require a real-time connection: monitoring the real world in real time, detecting any deviation between the simulation model and the real world, providing an approximate location for objects in the real world, checking for bottlenecks, etc.

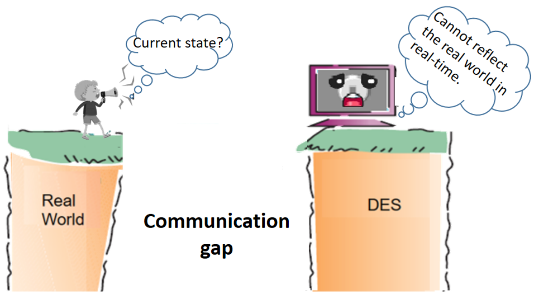


Figure 2.4: Connection problem with the real world in real time

2. DES uses historical data to predict the future (Tavakoli et al., 2008). However, using the simulation model to predict the future while depending on historical data could provide inaccurate prediction.

   DES models could be used to aid in the understanding of the behavior of the system and the system's functions as well as providing a low-cost, risk-free platform for testing various system configurations (what-if questions). A well-validated model of the system can also be used to anticipate future events. However, a DES which is based on historical data does not fulfill this credible exercise (Tavakoli et al., 2008).

   In health care in general and in the hospital in particular, there are many demands, variables, and factors that change continuously, and historical data cannot reflect these changes. To be more precise, Figure 2.5 depicts a time line divided into three periods: past, current, and future. The past is divided into two areas: Area 1 and Area 2. Area 1 represents all of the data or events that were collected from dates P1 to P2. Area 2 represents the data from P2 until now. If we need to predict the future based on the historical data from Area 1, and if we ignore the data that are in Area 2, we might have some inaccurate results. This is related to the fact that there can be many significant actions/events that have happened in Area 2, whereas these data did not exist in Area 1. In other words, it is possible that the prediction of the future based on the data from Area 1 could be suitable only for certain and particular situations that happened in the past.

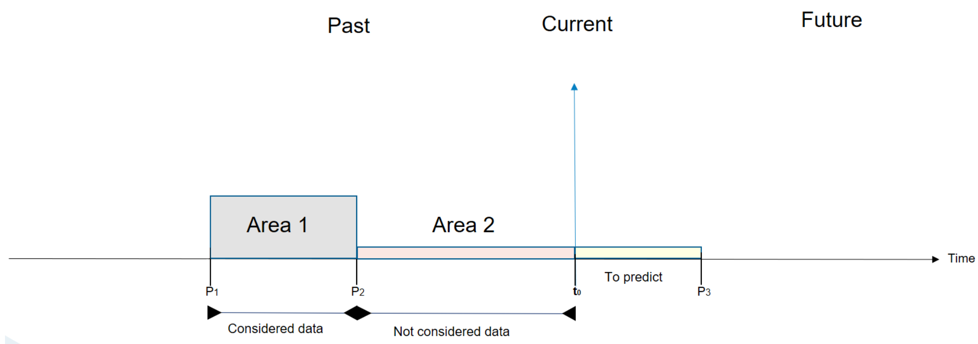


Figure 2.5: An inaccurate prediction for the future

3. DES commonly starts with an empty state (Hanisch et al., 2005). In fact, an offline simulation usually starts empty, then some parts (patients, for instance, in a healthcare simulation) enter the simulation. This means that starting the offline simulation does not reflect the current state of the real world. To solve this issue, a **warm-up period** is normally used to fill the offline simulation models with patients. After this warm-up period, the KPI can be collected from the model (the model will be in a steady state).

The question that may arise here is: “Will the state of the hospital be identical to the current state of the hospital after this warm-up period?”. Most likely, the answer is no. Warm-up periods work based on certain distributions, which is not the case in the real world. Figure 2.6 illustrates an example where the number of patients at 8:00 am after the warm-up period is 10, while in the real world the number of patients could be more or less.

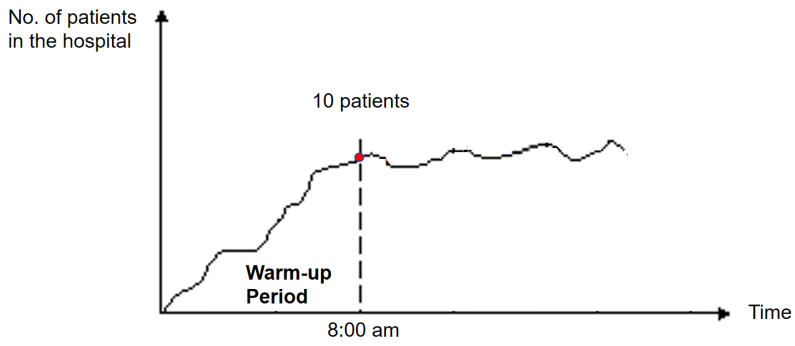


Figure 2.6: The warm-up period does not represent the current state

4. There is a limitation for running DES models. A DES cannot run to monitor the real world forever, which means that the simulation designer must decide when the simulation should finish. The typical choices are “at time t” or “after processing n number of events”. This is due to the fact that the simulation tools are designed to study the behavior of the system within a specific duration of time.

5. Offline simulation is considered to be time-consuming and costly (Tavakoli et al., 2008). This is because offline simulation usually involves the manual gathering, preparing, and processing of input data. In addition, working with such DES models requires experience, expense, and time for adjusting and calibrating these models. Furthermore, the cost of keeping a typical simulation model up to date and suitable for predictive analysis would be prohibitive due to the knowledge required and the time it takes to construct and run a good simulation.

Mitigating all of the above-mentioned limitations is part of the motivation for a new framework that can deal with DES, but in an online way to be used as a DT for monitoring the real world. This should support the connection with the real world: each time a new event is detected in the real world, this event must be reflected in the simulation tool. This section revealed the limitations for DES. In Section 2.4, we are going to talk about a new approach that could be used with DES to mitigate the aforementioned challenges.

2.4 Digital Twin Foundation

Global industries have undergone massive technological transformations due to the linkage between the digital and physical worlds. The use of Digital Twin (DT) technologies can help in this convergence.

The aim of this section is to provide answers to the following questions: (1) What exactly is a DT? (2) What is the role of a DT in the healthcare domain? (3) How can a DT be designed and implemented? (4) What are the most significant obstacles to the implementation of a DT?

2.4.1 Definitions

The concept of a "twin" goes back to 1970 with NASA's Apollo, where two exact copies of space vehicles were built. One was sent into space and the other stayed on the earth. The one that was on earth was referred to as the "twin". It was used to mirror the conditions of the space vehicle during the flight. The term "Digital Twin" appeared for the first time in 2003 in a presentation made by Dr. Michael Grieves (Grieves, 2003). In this presentation, Dr. Grieves presented the conceptual idea for Product Lifecycle Management (PLM), and included three main features for the DT: (1) physical space, (2) virtual space, and (3) a link between the data and the information that connect the two spaces.

Nowadays, in scientific research, many explanations and definitions can be found for the DT. For example, some researchers define a DT as a virtual representation that interacts with the physical system throughout its life cycle (Glaessgen et al., 2012; Grieves et al., 2017). Other widely used definitions emphasize the necessity for information to be exchanged between the real world and the virtual world, which includes sensors, data, and models. (Lee et al., 2013b; Negri et al., 2017). In other studies, a DT is considered the "cyber" segment of a Cyber-Physical System CPS (Alam et al., 2017; Graessler et al., 2017). Table 2.3 illustrates a list of the common definitions used in the literature for the DT.

Definition	Type of Digital Twin	Source
"A digital twin is an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin. The digital twin is ultra-realistic and may consider one or more important and interdependent vehicle systems, including propulsion/energy storage, avionics, life support, vehicle structure, thermal management/TPS, etc."	NASA Digital Twin	(Shafto et al., 2012)
"Ultra-realistic, cradle-to-grave computer model of an aircraft structure that is used to assess the aircraft's ability to meet mission requirements."	Airframe Digital Twin (ADT)	(Gockel et al., 2012)
"Coupled model of the real machine that operates in the cloud platform and simulates the health condition with an integrated knowledge from both data driven analytical algorithms as well as other available physical knowledge. It can also be described as a 5S systematic approach consisting of Sensing, Storage, Synchronization, Synthesis and Service."	Cyber-physical model	(Lee et al., 2013a)
"Very realistic models of the current state of the process and their own behavior in interaction with their environment in the real world."	Digital twins for the future of manufacturing	(Rosen et al., 2015)
"A high fidelity structural model that incorporates fatigue damage and presents a fairly complete digital counterpart of the actual structural system of interest."	Fatigue-damage model	(Bazilevs et al., 2015)
"Digital twin concept represents an innovative method to monitor and predict the performance of an aircraft's various subsystems. By creating ultra-realistic multi-physical computational models associated with each unique aircraft and combining them with known flight histories, operators could benefit from a real-time understanding of the vehicle's current capabilities." "Digital twin is a life management and certification paradigm whereby models and simulations consist of as-built vehicle state, as-experienced loads and environments, and other vehicle specific history to enable high-fidelity modeling of individual aerospace vehicles throughout their service lives."	Other DTs	(Bielefeldt et al., 2015; Hochhalter et al., 2014)

Note: +Common definitions from the literature, categorized by use.

Table 2.3: Different definitions of DT based on usage

From Table 2.3, it also appears that:

- Definitions of the DT may differ depending on the domain (e.g., aircraft factory, building construction, etc.)

- The notion of the DT can be applied to a product or a system (aircraft, vehicle), an operation (cyber-physical system) or a process (in manufacturing).

As a result, we propose a definition which is process oriented for the DT of hospital patient pathways.

Proposed definition for a DT of patient pathways in the hospital

A high-fidelity and dynamic virtual representation of the patient pathways in a hospital, created with data that is historically available. It is also designed to collect real-time data continuously from the real patient pathways. This DT is capable of providing information about the past states, the current states and the future states of patients.

In all of the research discussed in Section 2.4, the assumption is that the environment in which the DT is instanced is made up of three main components: the real world, the virtual world and the communication link between these two worlds. For us, the DT is considered to be a part of the virtual world that reflects the current state of a real-world environment (such as a hospital) in real time. This DT benefits from the real world through the communication link that updates its state and behaviors.

2.4.2 Applications for health care

Over the past few years, considerable literature has been published on the concept of the DT. This interest is mainly due to its ability to reduce the cost of system verification and testing in order to produce tailored decision support information and alerts for the users, and to predict changes in physical systems over time (Madni et al., 2019).

So far, the DT has been successfully implemented in a variety of industries, including manufacturing, aerospace, defense, manufacturing and building construction. These applications differ on the level of their focused functional description. Some of them are applied to the product/component level (such as for screening out unwanted product functionality and features), others focus on the operation level (such as for learning a new business practice or for assessing a new treatment), and others point out the process level (such as for predicting the near future of traffic).

Currently, various ongoing research demonstrates the importance of applying DT to the healthcare domain. (Martinez-Velazquez et al., 2019) suggest a human heart DT platform called “Cardio Twin”. The main idea of this twin might be summarized by collecting data from different sources, such as sensors, medical records and social networks. This data collection is handled using different techniques to detect whether the patient is suffering from heart disease, such as ischemic heart disease (IHD), or stroke. Another DT for the heart has been developed within the “Living Heart” project, promoted by the French software firm Dassault Systèmes. This consists of the first simulated real-life heart that serves as a common technological base for education and training, clinical diagnosis and prevention of heart disease. For instance, it can be used as a tool to guide device design and treatment planning of cardiac diseases such as stenosis and regurgitation, or prolapsing of the aortic, pulmonary, tricuspid or mitral valves (Baillargeon et al., 2014).

Within a broader scope, the Virtual Physiological Human (VPH), an EU project initiative, aims to develop an integrated model of human physiology at multiple scales, from the whole body through to the organ, tissue, cell and molecular levels and even down to the genomic level (Viceconti et al., 2008). The VPH is intended to support the development of patient-specific computer models and their application in personalized and predictive health care (Kohl et al., 2009). In the same vein, (Ayache, 2019; Ayache et al., 2011) propose a digital representation of the patient's anatomy and physiology based on executable models whose parameters can be automatically learned from real clinical, biological, behavioral and environmental data. The virtual patient can then be used to better quantify the observations, to simulate the evolution of a pathology, and to plan and simulate an intervention to optimize its effects. At another level, (El Saddik et al., 2019) present an ecosystem of the DT for health and well-being. This DT is capable of tracking and helping a person in case of an emergency, even if that person is alone and is suffering from heart diseases such as IHD. In a related matter, (Y. Liu et al., 2019) propose a cloud-based framework for elderly healthcare services using the digital twin, named CloudDTH. This DT framework aims to support the monitoring and real-time feedback for the elderly to manage their long-term lifecycle of health care. Machine-learning algorithms are implemented for fast simulations in order to predict crisis situations.

For a broader market sector, MyHealthAvatar is a EU research project attempting to create a digital representation of patient health status, called a digital avatar. This avatar plays a role similar to a personal digital health-related collection bag, carried by individual citizens throughout their lifetime and capable of sustaining all collected information in a meaningful manner. This information is related to multi-level personal health data that is collected from heterogeneous data sources such as clinical data, genetic data, and medical sensor data (Kondylakis et al., 2015). (Barricelli et al., 2020) propose an extension to SmartFit, which is a computational framework utilizing wearable sensors and Internet applications. The extension is able to monitor a team of athletes, predict their physical condition during training and then suggest changes in behavior to increase performance and health conditions.

From the healthcare organizational perspective, recent work carried out by the GE Healthcare company has implemented a predictive real-time platform in a new "command center" department for Johns Hopkins Hospital in Baltimore (INFORMS, 2017). It aims to predict patient activity and planning capacity according to demand, based on a DT of patient pathways and using prescriptive and predictive analytics, machine learning, natural language processing, and computer vision for better decision-making. These analytics provide hospital staff members with several accurate and timely insights on subjects such as bed assignments or whether a unit needs assistance or when an influx of patients is coming into the hospital. Similar work has also been pursued by the Siemens Healthineers company (Scharff, 2018), in which a medical unit DT was developed. It sets out to optimize the operational scenarios and layouts of a medical unit which may suffer from increasing patient demand, aging infrastructure and/or a lack of space by instantly evaluating various options before the implementation of the right solution to transform care delivery. This DT draws upon workflow simulation and a 3D computer model, enabling the building of a dynamic and comprehensive model which integrates patient pathways, staff scheduling and movements. Similarly, (Augusto et al., 2018) propose a framework for modeling and simulation in the form of an offline DT to evaluate the performance of emergency units in the case of a major crisis such as earthquakes, tsunamis or terrorist attacks. Table 2.4 provides a comparative study of these various DT applications according to their uses and their functional description level.

The current research backs up the effectiveness of Digital Twin Healthcare (DTH) in paving the road for patients to gain access to safe, effective new therapies and care while also lowering costs. However, as far as we know, and as various other studies on this topic have shown (Barricelli et al., 2019), DTH efforts are still partial, focusing mostly on the virtual patient and well-being. The role of the DT in hospital management, which is the focus of this research, has received little attention so far. Moreover, one of the hot topics that is considered to be relevant to the DT is the Cyber-Physical System (CPS). Section 2.4.3 illustrates the relation between the DT and the CPS in a simplified way.

Digital Twin project	Scope	Functional description level: Product	Functional description level: Operation	Functional description level: Process	Main usage: Learning	Main usage: Designing	Main usage: Monitoring	Main usage: Predicting	Main usage: Preventing
(Martinez-Velazquez et al., 2019)	Heart diseases	●	○	○	○	○	●	○	●
(Baillargeon et al., 2014)	Heart diseases	◐	●	○	●	●	○	●	○
(Viceconti et al., 2008), (Kohl et al., 2009)	VPH	●	●	○	●	○	○	○	●
(Ayache, 2019; Ayache et al., 2011)	Virtual patient	●	●	○	●	○	●	●	○
(El Saddik et al., 2019)	Health & well-being	●	○	○	○	○	●	●	●
(Y. Liu et al., 2019)	Health & well-being	●	○	○		○	●	○	●
(Kondylakis et al., 2015)	Health & well-being	●	○	○	○	○	●	●	●
(Barricelli et al., 2020)	Health & well-being	●	○	○	○	○	●	○	●
(INFORMS, 2017)	Medical unit organization	○	●	○	○	○	●	●	○
(Scharff, 2018)	Medical unit organization	○	●	○	○	●	○	○	○
(Augusto et al., 2018)	Medical unit organization	○	●	◐	○	○	●	○	●

○ = Not supported; ◐ = Partially supported; ● = Fully supported

Table 2.4: Comparing several DT projects dedicated to the healthcare sector

2.4.3 The relation between the cyber-physical system and the digital twin

The CPS is relevant technology, related to the DT, that has gained popularity in recent years for the purpose of improving overall system performance. CPSs "are systems engineered to integrate computational components, networking, and physical processes in a well-defined context to serve a specific purpose" (Arafsha et al., 2019): for example, in tracking the physical entity's movement for later processing and analysis. In fact, a comparison between the CPS and the DT is a contentious issue. Are they the same thing with different names, or are they different from a technical and application standpoint?

A hot topic among engineers and business stakeholders is this comparison between DT and CPS. In (Farsi et al., 2020), CPS is another term for the DT phenomenon, where the physical system acquires sensory data from the real world and communicates it to the computational modules of the CPS via communication technologies (wireless). This allows for easier data analysis based on the control of the resources or physical environments. In addition, (Tao et al., 2019) stated that CPSs and DTs have similar characteristics, and both describe the merger of the cyber and physical worlds, but they are not identical; the sensors and actuators are considered core elements in CPS, while models and data are the core elements in a DT. For some researchers, such as (Leng et al., 2019), the DT is considered to be the cyber part of the CPS; a CPS is a set of physical devices that communicate with a virtual cyberspace via a communication network. Each physical device will have a cyber part, which will be a digital representation of the real device, resulting in DT models. As a result, the DT can monitor and control the physical entity, while the physical entity can provide data to keep its virtual model up to date and synchronized. For (Steinmetz et al., 2018), the DT is one of the most crucial ideas in the CPS era. It can bring benefits and advantages such as simulation, monitoring or management once it connects the physical and the virtual over the IoT. Last but not least, for (Arafsha et al., 2019) a DT system is a certain type of CPS where the components and the attributes of the physical systems are mirrored in the cyber world.

Based on different opinions from the state of the art, by doing mapping between the CPS and DT architectures, we claim that the virtual world is equivalent to the cyber space in the CPS where the DT exists. The real world is synonymous with the physical space in the CPS where the real sensory entity exists. Furthermore, the connection between the cyber and physical spaces correspond to the connection link (communication environment) between the virtual and the real world, as illustrated in Figure 2.7.

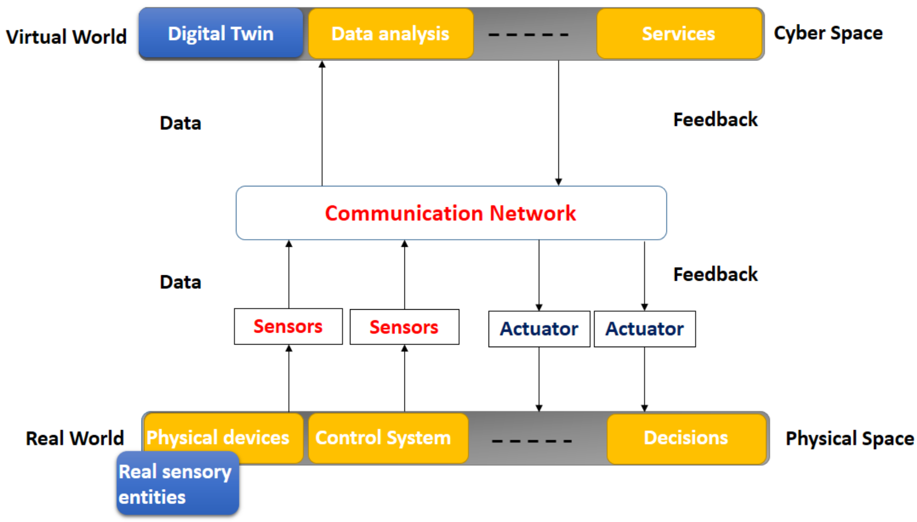


Figure 2.7: Simplified structure for the Cyber-Physical System (CPS)

More details about the different layers presented in Figure 2.7: the real world, the virtual world, and the connection link between them will be clarified in Section 2.4.5. In addition, different technologies that make that connection between the real world and the virtual world possible will be discussed in Section 2.4.7. However, according to the relation between these layers, the DT can be classified into two types based on loop of benefits: closed-loop DT and open-loop DT. More details about these loops will be explained in Section 2.4.4.

2.4.4 The Loop of Benefits

In the literature, there are two types of DTs based on the control feedback called the loop of benefits : closed-loop DT and open-loop DT (Aydt et al., 2009).

In a closed-loop DT, there is control feedback from the simulation model in the virtual world to the physical system in the real world, whereas in an open-loop DT, there is no such feedback, given that the simulation model has no effect on the physical system. For example, weather forecasting applications can use DTs for predicting and validating models, but (unfortunately) not for controlling the real world, as the weather cannot be affected or controlled by an actuator. Therefore, it is not possible to build a weather control system.

However, the idea of a closed-loop DT came from symbiotic simulation, "which refers to a close relationship between a simulation system and a physical system" (Fujimoto et al., 2002). This was originally defined at the Dagstuhl seminar on Grand Challenges for Modeling and Simulation in 2002, where the speaker stated that: "The simulation system benefits from real-time measurements about the physical system which are provided by corresponding sensors. The physical system, on the other side, may benefit from the effects of decisions made by the simulation system".

In the closed-loop DT, two types of control feedback exist (Aydt et al., 2009): direct control feedback, and indirect control feedback. In direct control, the DT can affect the physical system directly through actuators, and this effect could be positively or negatively based on the accuracy of the DT. In indirect control, the DT can affect the physical system through decision-makers. In this case, the accuracy of the decision depends on the accuracy of DT.

In the context of online simulation and symbiotic simulation, various classes of DT exist, depending on whether the DT is open loop or closed loop (direct or indirect control). These classes are illustrated in Table 2.5.

Class	Purpose	Feedback loop
DT control system	Control of a real system	Closed-loop (Direct control)
DT decision support system	Support the decision-maker	Closed-loop (Indirect control)
DT forecasting system	Forecast and predict the near future of the real system	Open-loop
DT model validation system	Validate the simulation system	Open-loop
DT anomaly detection system	Detect anomalies either in the real system or in the simulation model	Open-loop

Table 2.5: Classes of DT based on feedback loop

Table 2.5 illustrates five classes of DT, depending on the feedback loop: (1) a DT control system, (2) a DT decision support system, (3) a DT forecasting system, (4) a DT model validation system, and (5) a DT anomaly detection system. The first two classes are considered closed-loop DTs because there are control feedback returns from the DT to the real world, whether this feedback is direct, which means it can control the real world using actuators, or indirect, like alarm messages, which return to the decision-maker and he/she can make a decision to manage and/or control something in the real world. The last three classes are considered open-loop DTs because there is no control feedback returning from the DT to affect the real world.

In this research work, we are going to focus on the second class, which is the DT decision support system. This DT will be used to aid the decision-maker and/or the hospital manager to: (1) monitor the patient pathways in the hospital, (2) predict the near future of these pathways, (3) manage and control these pathways based on the alarm messages that return from the Digital Twin for Monitoring (DTM), which will be discussed in detail in Chapter 3.

2.4.5 Characteristics: the real world, the virtual world and their communication

Various characteristics can be identified in definitions of the DT. For example, ultra-realistic, multi-physical, computational models, etc. To understand the characteristics (features) of the DT, this section aims to investigate and answer the following three questions:

- What are the main characteristics of the real world?
- What are the main characteristics of the virtual world?
- What are the main characteristics of the communication environment that connects the virtual world to the real world?

For the sake of clarity and to address the above questions and make the concept more understandable, we will utilize the following definitions:

> For the sake of clarity and to address the above questions and make the concept more understandable, we will utilize the following definitions:
>
> - Real world: the environment where the physical entity (real sensory objects) exists. For example, a real hospital.
> - Virtual world: the environment where the virtual entities exist. For example, the virtual representation of a hospital.
> - Real sensory object: a living or non-living object that has sensing capability, thanks to sensors that can capture/collect and send near real-time data. For example, a real patient is a living entity; a real wheelchair is a non-living entity. They are both sensory if they each have a least one sensor that sends data.
> - Virtual entity: a representation of a living or non-living object. For example, a virtual patient is a living entity; a virtual wheelchair is a non-living entity.

As far as we know, several research works outline the main features of a DT. Among them, (El Saddik, 2018) has defined seven features for the DT: (1) unique identifiers, (2) sensors and actuators, (3) artificial intelligence, (4) communication, (5) representation, (6) trust, and (7) privacy and security. (Jones et al., 2020) conducted a systematic literature review to identify the characteristics of the DT. As a result, they identified twelve features for the DT, as follows: (1) a physical entity, (2) a virtual entity, (3) a physical environment, (4) a virtual environment, (5) fidelity, (6) a state, (7) parameters, (8) a physical-to-virtual connection, (9) a virtual-to-physical connection, (10) twinning and a twinning rate, (11) physical processes, and (12) virtual processes. Furthermore, (Rodríguez-Aguilar et al., 2020) presented thirteen features for the DT as follows: (1) a physical entity/ twin, (2) a virtual entity/twin, (3) a physical environment, (4) a virtual environment, (5) a state, (6) metrology, (7) realization, (8) twinning, (9) a twinning rate, (10) a physical-to-virtual connection, (11) a virtual-to-physical connection (12) physical processes, and (13) virtual processes. Moreover, seven features of the DT have been defined by (Singh et al., 2021): (1) high-fidelity, (2) dynamic, (3) self-evolving, (4) identifiable, (5) multi-scale and multi-physical (6) multidisciplinary, and (7) hierarchical. (Fei et al., 2018) stated different features for DTs, such as: (1) ultra-high synchronization (2) fidelity, and (3) convergence between the physical and virtual. (Z. Zhu et al., 2019) summarized the DT features as follows: (1) ultra-high synchronization, (2) fidelity, (3) reflecting the physical part in real time, (4) interaction and convergence, and (5) self-evolution. (VanDerHorn et al., 2021) characterized the DT by three main components: (1) physical reality, (2) virtual representation, and (3) interconnections that exchange information between the physical reality and virtual representation. Last but not least, (Wang, 2020) characterized the DT by the following features: (1) uniqueness, (2) multi-physical, (3) multi-scale, (4) hierarchical, (5) integrated, (6) dynamic, (7) super-realistic, (8) computability, (9) probability, and (10) multidisciplinary.

Figure 2.8 shows a network virtualization for the density of DT characteristics according to these references, while Table 2.6 provides a description of these characteristics according to the different researchers.

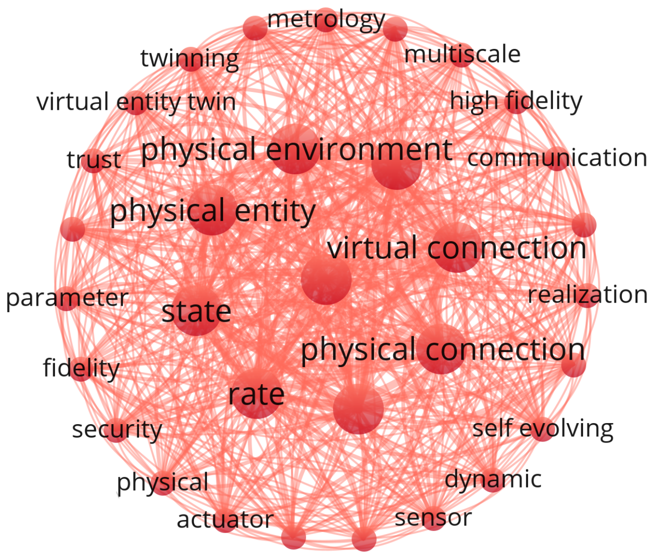


Figure 2.8: Density virtualization for digital twin characteristics

As is obvious from the above-mentioned DT features, most of the research works made no distinction between the DT's characteristics and those of the real world or their communication environment. The reason behind this may be that for them, the DT includes all three components together. To clarify this, questions from the engineering point of view could be asked regarding the DT's characterizations. These questions may help analyze, along with clustering, the aforementioned characterizations according to their existence in the real world, the virtual world, and their connection environment. These questions are illustrated in Table 2.7.

Based on the characteristics in Table 2.7, we have classified these characteristics according to their existence in the real world, the virtual world, and the communication environment, as illustrated in Table 2.8.

As shown in Table 2.8, several features can exist in a virtual world; for example, the models will be used to represent the real sensory objects in the virtual world corresponding to virtual entities. Each virtual entity must reflect one real sensory object. The virtual entity must be characterized by the high-fidelity feature, which means the virtual entity must be as close as possible to this real sensory object in terms of visualization, behaviors, etc. Furthermore, the virtual world must have an application programming interface for receiving data from the real world as well as for sending feedback and information to this real world. Sometimes, different types of DTs are required to collaborate with each other. In this case, the inter-operability feature must be achieved by the DT. However, the virtual world where the DT exists must be synchronized with the real world where the real sensory object exists, thanks to the different types of data and knowledge that are exchanged between the virtual world and the real world in near real time. Cybersecurity algorithms and mechanisms must be applied to protect this data. In case of an unexpected event happening in the real world, the DT must be intelligent enough to provide a quick response to the real world.

In the real world, each real sensory object must be attached to a unique identifier (ID) that distinguishes which virtual entity corresponds to this real sensory object. Different Internet of Things (IoT) sensors could be used in the real world. These sensors will facilitate the convergence between the real world and the virtual world by capturing different data about the real sensory object to feed its corresponding virtual entity. Different types of resources must be present in the real world to facilitate a quick response to the feedback that comes from the virtual world in order to mitigate, reduce or recover from any unexpected event. Cybersecurity must be deployed at different levels in the real world, starting from securing the physical equipment and extending to the exchange of data/knowledge between the real world and the virtual world.

In the communication environments, different features must be applied. For example, the communication environment must be able to allow the data and information to be exchanged between the real world and the virtual world in real time. This will allow for a timely synchronization between the virtual world and the real world, thanks to the different network protocols, data storage capabilities, etc. This environment must allow the different DTs in the same or in different domains to interact with each other. Because this environment is considered the heart of the communication between the virtual world and the real world, it must be as secure as possible. To do this, different security techniques could be used. In fact, hacking this environment means hacking the virtual world as well as hacking the real world.

> In Table 2.8, the characterizations that can be applied to the different layers (real world, virtual world, and the communication environment) can been seen. The characteristic of any one feature could be different, depending on which layer it is used in. For example, in the virtual world, agility refers to the ability of the DT to respond quickly in different situations, such as in the case of an unexpected event. In the real world, the decision-maker must be able to make a quick decision according to the feedback received from the DT. Moreover, the communication environment must be able to exchange the data and information between the real world and the virtual world as quickly as possible. Another example is security, which could be adapted to all three layers, but with using different mechanisms. As for the communication link, it refers to all of the components from the real world, the virtual world and their communication environment that can be used to connect the virtual with the real world. Based on the required characterization, each layer could have different components to make the connection between the two worlds possible. Conversely, features can be applied to one or to the two layers, as depicted in Table 2.8.

Worthwhile questions

On the subject of security, what is the relation between the security and the performance of the DT? How can we measure the security of the DT? All of these questions and others can be considered open questions to be answered in the future.

The main and common characteristics of the virtual world, the real world and the communication environment have been depicted in this section. These features may be altered depending on the application domain, the topic under investigation, and the decision-making objectives. For instance, investigating the activity of the human heart may require characteristics and parameters to be as similar to the real world as possible. In our case, monitoring patients in pathways may require a less precise characterization. However, most of the time it is difficult to achieve the aforementioned features of a DT. The reason behind these difficulties originates in a complex physical system. For example, the human body is not easily represented as a set of variables and attributes. This return to the technology used, such as the IoT, can help in quantifying part of the aspects of the physical object. Developing a virtual replica for a sensory twin is based on the ability of software environments to virtualize components of this twin. Moreover, the information interchange between the real sensory twin and the DT should be timely; that is, the time between real sensory twin state changes should be short in relation to the needs and intended usage of the DT. This could depend on the quality and performance of the technology used.

Characteristic	Description	Source
Physical environment	The real world environment where the physical entity exists.	(Rodríguez-Aguilar et al., 2020) (Jones et al., 2020)
Virtual environment	Any number of virtual worlds or simulations that mimic the status of the physical environment and are designed for certain use-cases, such as health monitoring and production schedule optimization.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Parameters	"Any of a set of physical properties whose values determine the characteristics or behavior of something parameters of the atmosphere such as temperature, pressure, and density." (Merriam-Webster, 2021).	(Jones et al., 2020)
Fidelity	This refers to how elaborately, specifically, and exactly the virtual entity conforms to the real structure, behaviors, and personality of its real entity. In other words, the number of parameters transferred between the physical and virtual entities, as well as their precision, and their level of abstraction.	(Fei et al., 2018) (Z. Zhu et al., 2019) (Jones et al., 2020) (Singh et al., 2021)
State	The current value of all parameters of the physical or virtual entity or the physical or virtual environment.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Physical-to-virtual connection	The connection from the physical to the virtual environment. This reflects the extensive process of connecting various parts of a physical entity to a virtual entity.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Virtual-to-physical connection	The connection from the virtual to the physical environment.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Twinning and the twinning rate	The act of synchronization between the two entities, as well as the rate at which it happens.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Physical processes	The physical process and processes within which the physical entity involved, such as a manufacturing production line.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Virtual processes	The computational techniques used within the virtual world. For example, optimization, prediction, simulation, analysis, integrated multi-physics, multi-scale, probabilistic simulation.	(Jones et al., 2020) (Rodríguez-Aguilar et al., 2020)
Metrology	The act of measuring the state of the physical/virtual entity/twin. For example, sensors relaying key measures from the real world to the virtual world.	(Rodríguez-Aguilar et al., 2020)
Realization	The act of changing the state of the physical/virtual entity/twin.	(Rodríguez-Aguilar et al., 2020)
Dynamic	The physical system is considered dynamic, which means that it changes over time. As a result, a DT must also change as the physical system changes. In other words, the DT must mirror the physical system and its behavior realistically in the virtual world.	(Wang, 2020) (Singh et al., 2021)
Self-evolving	Throughout its life cycle, a DT evolves alongside its physical counterpart. Any changes made to either the physical or digital twin are mirrored in the counterpart, resulting in a closed feedback loop.	(Z. Zhu et al., 2019) (Singh et al., 2021)
Identifiable / unique identifier	Every physical asset requires its own DT.	(Wang, 2020) (El Saddik, 2018) (Singh et al., 2021)
Multi-scale (geometric properties)	The DT contains a collection of information about the physical twin ranging from the micro-atomic to the macro-geometric level, such as shape, size, tolerance, and so on, as well as on microscopic properties such as surface roughness, etc.	(Wang, 2020) (Singh et al., 2021)
Multi-physical (physical properties)	The DT models are also based on physical properties (structural dynamic models, stress analysis models, etc. and material properties of physical twins, such as stiffness, strength, etc.)	(Wang, 2020) (Singh et al., 2021)
Multidisciplinary	The DT revolves around many disciplines, such as computer science, information technology, communications and mechanics, etc.	(Wang, 2020) (Singh et al., 2021)
Hierarchical	A DT can be viewed as a collection of interconnected sub-models. For example, the DT of an aircraft is comprised of a rack DT, a DT of the flight control system, a DT of the propulsion system, etc.	(Wang, 2020) (Singh et al., 2021)
Communication	DTs should be able to interact and communicate with the real twins, and/or other DTs in near real time.	(El Saddik, 2018)
Representation	Depending on the application, DTs could have a virtual representation, such as a 3D model.	(El Saddik, 2018) (VanDerHorn et al., 2021)
Trust	The real sensory object must ensure that its virtual entity can be relied on to conduct sensitive responsibilities like handling financial transactions or a stock portfolio for the real sensory object, or interacting on behalf of the real sensory object in meetings.	(El Saddik, 2018)
Privacy and security	DTs should be able to preserve their real twin's identity and privacy.	(El Saddik, 2018)
Computability	Simulate and reflect the status and behavior of the corresponding physical entity in real time.	(Wang, 2020)
Probability	DT models enable computation and simulation using probabilistic statistics.	(Wang, 2020)
Integrated	Multiple physical structural models, geometric models, and material models are combined into a multi-layer and multi-scale integrated model.	(Wang, 2020)

Table 2.6: Descriptions for DT characteristics according to different research works

Engineering points of view	Characterization	Components	(El Saddik, 2018)	(Fei et al., 2018)	(Zhu et al., 2019)	(Jones et al., 2020)	(Rodríguez-Aguilar et al., 2020)	(Wang, 2020)	(Singh et al., 2021)	(VanDerHorn et al., 2021)
How is the real sensory object represented as a virtual twin?	Modeling	3D models, mathematical models, etc.	X	X	X	X	X	X	X	X
How to distinguish which virtual twin corresponds to which real sensory object?	Identifiability	Tags, where each tag has a unique identifier.	X					X	X	
What are the different parameters that must be considered to make the behaviors, the structure, and the visuals of the virtual twin close to the real sensory object?	Fidelity	- Number of parameters, entities, processes, states, behaviors, functions, scales. - The type of physical properties.		X	X	X	X	X	X	X
How will the information and data be exchanged between the real world and the virtual world?	Link	- Physical-to-virtual connection. - Virtual-to-physical connection. - IoT devices, actuators, data storage. - Application programming interface.	X	X	X	X	X	X	X	X
How do different digital twins interact with each other?	Federation	Interoperability, hierarchical, multidisciplinary.						X	X	
How can the digital twin mimic the real sensory twin?	Synchronization	Twinning, self-evolving, dynamic, etc.	X	X	X	X	X		X	X
How can the digital twin make a quick response according to real world events/ opportunities?	Agility	Resources (infrastructure, hardware, software, human, etc.) , plan strategy, up-to-date platform, etc.	X	X	X			X		X
How much time will the information/data take to travel from the real world to the virtual world and back?	Latency	Twinning rate (near real time).	X	X	X	X	X			X
How can the digital twin make an intelligent decision related to the real world?	Intelligence	Artificial intelligence, ontologies, machine learning, deep learning, etc.	X							X
How can the digital twin keep real-world data as secure as possible? And how will the data be transmitted in a secure way between the real world and the virtual world and vice versa?	Cybersecurity	Security pillars: confidentiality, integrity, and availability.	X			X				
How can the real world and the virtual world benefit from each other?	Desired benefits	Data exchanging, monitoring, predicting, optimizing, validation, etc.			X	X				X

Table 2.7: A comprehensive view of DT characteristics

#	Characterization	Real world	Virtual world	Communication environment	Description
1	Models		X		Different types of models could be used to represent the real sensory objects as well as their structure, behavior, etc.
2	Identifiable	X	X		Unique IDs could be used with real sensory objects in order to communicate with their virtual entities and vice versa.
3	Fidelity		X		The virtual world must include all of the required information that makes the virtual object as close as possible to its real sensory object twin. Likewise, the virtual environment shall be as close as possible to the real world environment. The precision of the fidelity could depend on the analytical perspectives of the characterization of the real sensory object. For example, the more the characteristics of the real sensory objects are reflected in the virtual world, the higher the fidelity will be.
4	Link	X	X	X	The link refers to the different components (hardware or software) used to easily make the connection between the real world and the virtual world.
5	Federation		X	X	Refers to a collaborative method or mechanism that allows two or more individual DTs in different domains or the same domain to work together. The inter-operability of various IoT platforms could be used to achieve federation technologies.
6	Synchronization		X		The act of synchronizing the virtual twin with its real sensory object.
7	Agility	X	X	X	The ability to respond quickly to real-time changes.
8	Latency			X	To achieve synchronization between the virtual world and the real world at near real time, and with a high-fidelity visualization, the latency time must be as short as possible.
9	Intelligence		X		The DT must be intelligent to make quick and intelligent decisions.
10	Cybersecurity	X	X	X	Different levels of security shall be applied to the real world, the virtual world and the connection to keep the privacy and trust of the DT.
11	Desired benefits	X	X		The real world and the virtual world can benefit from each other through the communication link.

Table 2.8: Characteristics of the real world, the virtual world, and the communication link

2.4.6 A digital twin architecture reference model

Until now, there has been no common, clear or standard approach to building a generic DT (Tao et al., 2018a). Each designer proposed their own way of building it. Several types of architectures have been reported in the literature for addressing this issue. These architectural types could be classified by their number of layers: two-layer architecture, three-layer architecture, four-layer architecture, five-layer architecture, and six-layer architecture. The first architecture dates back to (Grieves, 2014). In this architecture, three main parts (layers) can be used to develop the DT: (1) physical space, (2) virtual space, and (3) the connection between these spaces. In the second architecture, it is a five-layer DT proposed by (Ponomarev et al., 2017): (1) a cyber-physical layer, (2) a primary processing/store data layer, (3) a distributed computing and storage layer, (4) a model and algorithm layer, and (5) a visualisation and user interface layer. In the third architecture, another five-layer DT architecture is proposed and developed by (Tao et al., 2018b): (1) the physical entity, (2) the virtual entity, (3) the services for physical and virtual entities, (4) the data, and (5) the connection between all of the above-mentioned layers. Moreover, there is an extended six-layer DT, which was developed by (Redelinghuys et al., 2018). The layers are: (1) physical devices, (2) local controllers, (3) local data repositories, (4) an IoT gateway, (5) cloud-based information repositories, and (6) emulation and simulation. Furthermore, (Leng et al., 2019) suggested four layers: (1) process, (2) physical, (3) cyber, and (4) social. (Aheleroff et al., 2021) introduced five layers: (1) an application layer, (2) a cyber layer, (3) a digital layer, (4) communication, and (5) a physical layer. A two-layer architecture is proposed by (Talkhestani et al., 2019). In this architecture, there are (1) a cyber layer and (2) a physical layer. (Bevilacqua et al., 2020) described four layers: (1) a process industry physical space, (2) a communication system, (3) a digital twin, and (4) a user space, whereas (Danilczyk et al., 2019) talked about two-layer architecture: (1) a digital twin, and (2) a physical twin. Another four-layer architecture is proposed by (Gehrmann et al., 2019) : (1) a virtual domain, (2) an intermediate network, (3) a physical domain, and (4) an external network. Additionally, (Zheng et al., 2019) proposed three layers: (1) virtual space, (2) an information processing layer, and (3) physical space. (Wu et al., 2021) specified five layers (dimensions) that are represented by (1) the physical entity, (2) the virtual entity, (3) the services module, (4) the digital twin data module, and (5) the connection module. A three-layer architecture is presented by (Qi et al., 2018): (1) a unit level, (2) a system level, and (3) a system of systems level. Finally, three layers are illustrated by (Lin et al., 2019): (1) an operation layer, (2) a visualization layer, and (3) an intelligence layer. All of these layers are organized in Table 2.9.

Based on the aforementioned DT architecture layers, it is worth noting that there are numerous views for the architecture of the DT. Table 2.10 illustrates three levels of views, where each level illustrates a different degree of detail. For example, the high-level view shows three layers for the DT architecture, such as the real world, connection, and the virtual world, while the mid-level view shows different details for each high-level layer, such as algorithms, decisions, models, etc, without clarifying them. The last level, which is the deep-level view, details each layer that exists in the mid-level. For example, in the mid-level, we know that there are different kinds of algorithms that could be used with DT. In the deep-level view, the details of these algorithms exist; for example, initialization algorithms, synchronization algorithms, and so on.

(Grieves, 2014) illustrates the high-level view for the main parts of the DT that includes real space, virtual space, and the connection between them, whereas (Ponomarev et al., 2017) merges the real space and the virtual space into a single layer known as the cyber-physical layer. Furthermore, they focus their architecture on distributed big-data storage management. For us, this architecture could be considered a mid-level view of the DT architecture because there are some details regarding the storing and processing of the data. In (Tao et al., 2018b), a mid-level view of the DT architecture is proposed. In this view, they separate the data in

Source	Two-layers	Three-layers	Four-layers	Five-layers	Six-layers	Layers
Grieves, 2014		X				(1) physical space (2) virtual space (3) connection
Ponomarev et al., 2017				X		(1) cyber-physical layer (2) primary processing/storage data layer (3) distributed computing and storage layer (4) models and algorithms layer (5) visualisation and user interfaces layer
Tao et al., 2018b				X		(1) physical entity (2) virtual entity (3) services for physical and virtual entities (4) data (5) connection
Redelinghuys et al., 2018					X	(1) physical devices (2) local controllers (3) local data repositories (4) IoT gateway (5) cloud-based information repositories (6) emulation and simulation
Leng et al., 2019			X			(1) process (2) physical (3) cyber (4) social
Aheleroff et al., 2021				X		(1) application layer (2) cyber layer (3) digital layer (4) communication (5) physical layer
Talkhestani et al., 2019	X					(1) cyber layer (2) physical layer
Bevilacqua et al., 2020			X			(1) process industry physical space (2) communication system (3) digital twin (4) user space
Danilczyk et al., 2019	X					(1) digital twin (2) physical twin
Gehrmann et al., 2019			X			(1) virtual domain (2) intermediate network (3) physical domain (4) external network
Zheng et al., 2019		X				(1) virtual space (2) information processing layer (3) physical space
Wu et al., 2021				X		(1) physical entity (2) virtual entity (3) services module, (4) digital twin data module (5) connection module
Qi et al., 2018		X				(1) unit level (2) system level (3) system of systems level
Lin et al., 2019		X				(1) operation layer (2) visualization Layer (3) intelligence layer

Table 2.9: Architectural layers for designing a DT

one layer as well as mention a new layer which is called the service layer. Moreover, they replace the physical and the virtual spaces to use entities. As far as we know, real entities could be a subset of the real space and virtual entities could be a subset of the virtual space. In the same vein, (Redelinghuys et al., 2018) presents a deep view of details of the architecture of DT. For example, they focus on the technologies (hardware and software) that will be used to collect the data from the physical devices to feed the simulation and the emulation tools, and how the information will return to the real world again. They mention the IoT and in addition, they mention the local controllers, which are usually used to control the physical devices based on the control feedback that may be received from the virtual world. They identify the technology of the virtualization and representation, which is simulation and emulation. (Leng et al., 2019) replace the virtual world with the cyber world, and they include the social layer which includes people, devices, etc. This layer is considered a mid-level view. (Aheleroff et al., 2021) presents a mid-level view. In this architecture, an application layer is included to represent the different types of data, knowledge, and wisdom. (Talkhestani et al., 2019) presents a very high-level view for the DT architecture, where they merge the storage, the data, algorithms, etc. in the cyber layer. (Bevilacqua et al., 2020) shows a mid-level view. Furthermore, (Danilczyk et al., 2019) shows a high-level view architecture. Last but not least, (Gehrmann et al., 2019) illustrates a mid-level view. In this architecture, the external network is used instead of using a social layer, as in (Leng et al., 2019).

Digital Twin Architectural Layout			Grieves, 2014	Ponomarev et al., 2017	Tao et al., 2018b	Redelinghuys et al., 2018	Leng et al., 2019	Aheleroff et al., 2021	Talkhestani et al., 2019	Bevilacqua et al.,2020	Danilczyk et al., 2019	Gehrmann et al., 2019
High-Level	Mid-Level	Deep-Level										
Real World	Physical Sensory entity	1. Living entities	X	X	X	X	X	X	X	X	X	X
		2. Non-living entities	X	X	X	X	X	X	X	X	X	X
	Controllers	1. Process	X	X	X	X	X	X	X	X		
		2. Resources	X	X	X	X	X	X			X	
		3. Other							X		X	
	Decisions	1. Plan strategy		X	X		X			X		
		2. Resources	X	X		X		X			X	
		3. Other						X				
	KPI	1. Main objective		X	X							X
		2. Fault/failure		X			X					
	Security	Physical attack				X		X			X	X
Connection	Data processing	1. Cloud Computing		X	X	X	X	X	X	X	X	
	Knowledge	2. Big data		X		X	X	X				
	Communication network	3. IoT devices	X	X		X		X	X	X	X	
	Security	Network security				X		X		X	X	X
Virtual World	Virtual entity	1. Living entities	X	X	X	X	X	X	X	X	X	X
		2. Non-living entities	X	X	X	X	X	X	X	X	X	X
	Algorithms	1. Synchronization	X			X	X		X	X		X
		2. Initialization			X	X	X					X
		3. Artificial Intelligence		X		X			X	X	X	
		4. Optimization algorithms				X	X			X		
	Models	1. Mathematical models		X	X					X		
		2. 3D models			X				X	X		
	Services	1. Monitoring	X	X	X	X	X	X	X	X	X	X
		2. Predicting	X		X	X	X	X	X	X	X	
		3. Optimization		X	X	X	X	X	X	X		
		4. Maintenance	X		X	X	X	X			X	X
		5. Security				X					X	X
	Virtualization and representation	1. Simulation	X		X	X	X	X	X	X	X	X
		2. Emulation				X						X
		3. Other		X								
	Feedback	1. Alarms		X						X		
		2. Unexpected events									X	
		3. Failures/other			X			X	X		X	

Table 2.10: Proposed architectural layers for designing a DT

In this section we have noticed that the architecture of the DT depends more or less on the level of detail for some of the services, the technological aspects in terms of hardware and software, etc. However, all of these approaches have the same goal, which are the main layers that are required to design and develop the DT reference model even if they have different levels of detail.

In our research work, we will consider the DT to be an artifact that exists in one layer of the high-level architecture, namely the virtual world, as depicted in Figure 2.9. In other words, the DT is represented in the virtual world and can communicate with the real world through a connection link, as demonstrated in the mid-level architecture in Figure 2.10. This architecture includes the real world (real space), the virtual world (virtual space), and the communication between them (connection). This architecture illustrates a mid-level view for the virtual world. To serve each level, different technologies and services can be deployed. Some developers, for example, will utilize a simulation tool, while others may combine a supervisory tool with an ML tool. Additionally, some developers concentrate solely on local processing, while others concentrate on both local and cloud processing. Some developers may want to integrate security in all three layers to secure the DT, while others may want to integrate security in just two layers, whereas others do not care. Developers and designers of DTs may have various perspectives on the mid-level and deep-level details.

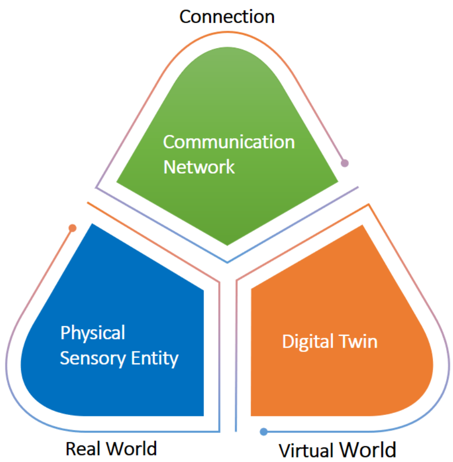


Figure 2.9: High-level architecture showing the DT concept

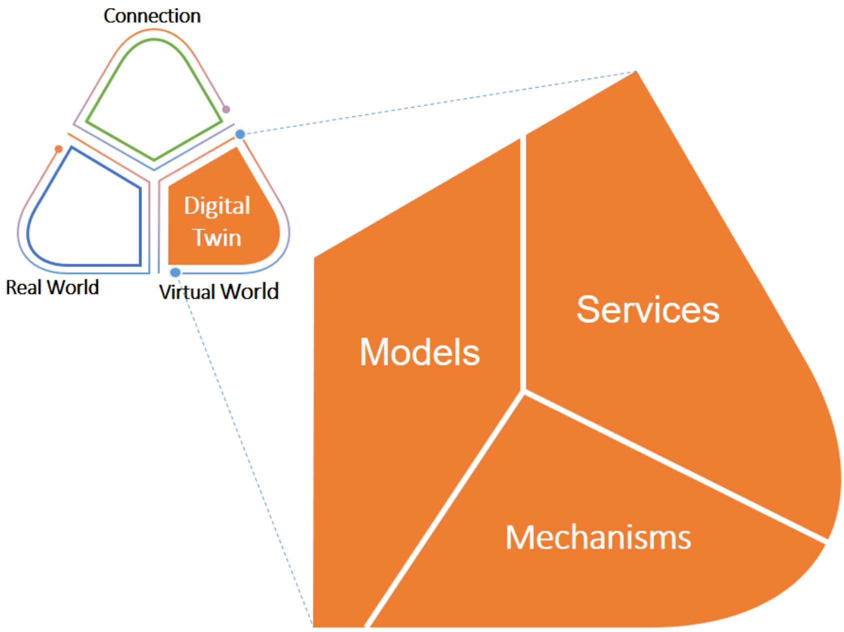


Figure 2.10: Mid-level architecture showing different virtual world components among others

According to the survey on DTs done by (Barricelli et al., 2019), there are two different methodologies for constructing a DT. The first approach focuses on constructing a DT for a physical entity that has yet to be twinned. The second approach concerns the creation of a DT for a physical entity that already exists but has no DT in place. Thus, we have four different views related to the real-world layer (physical layer/ physical space layer/ real space layer), as follows:

1. Building a DT for a system that does not exist. This is considered a type of offline simulation (traditional simulation), as explained in Section 2.3, and in this case, the real world is not part of the DT.

2. Building a DT to simulate the behavior of an existing system using historical data and starting in an empty state. In this case, the DT is a type of offline simulation (prospective simulation), where the real world is not a part of DT; what existed in the past may not be suitable for today.

3. Building a DT to predict the behavior of an existing system, where this simulation is initialized by the current state of the real world, as well as being run on the latest up-to-date dynamic distributions. This is for us called an offline DT (semi-online simulation), which means the DT is connected with the real world for two reasons: (1) initializing the simulation model with the current state of the real world, and (2) updating the distributions used in the simulation model to include the latest data, then continuing to work offline. The real world is not part of the DT, but the DT benefits from the real world to have “fresh” data.

4. Building a DT for an existing system and there is a communication link between the DT and the existing system, which means any changes in the real world will be reflected

in the simulation model. For us, this is called a DT (or an online DT). We benefit from the real world to run this DT, but that does not mean the real world is part of the DT. Stopping the DT will not stop the real world, but stopping the real world could lead to stopping the DT.

In this work, we will consider the third and the fourth type of DT (offline DT and online DT). The DTM which will be used in this work to monitor the patient pathways in real time is an online DT, whereas the Digital Twin for Predicting (DTP) that will be used in this work to predict the near future of the pathways based on current data is an offline DT (predictive DT). Furthermore, in our opinion, depending on the type of service, the service component could be included in all three of the layers (real world, virtual world, and the connection layers). For example, some services could be used to share the DT with more than one organization. Here, this service component could be included in the connection layer. Sometimes, the service component would be used for monitoring, predicting or optimizing the real system. In this case, the service component could exist in the virtual world layer, and so on. A deep-level view for the proposed DT is illustrated in Figure 2.11.

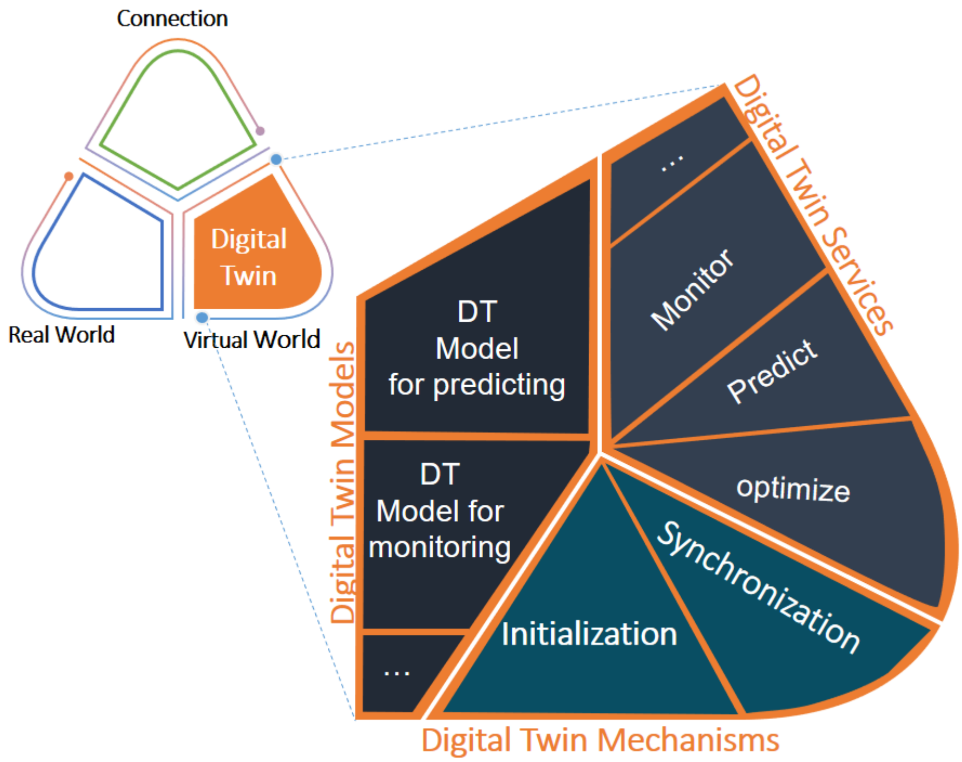


Figure 2.11: Deep-level view for the proposed digital twin

In Figure 2.11, three layers are illustrated: (1) the real world, where the real sensory object exists, (2) the virtual world, where the DT exists, and (3) the connection link, where the communication network exists. This figure shows that there are different components related to the DT, such as: DT services, DT models, DT algorithms (mechanisms), etc. All of these layers, in addition to some of their components, will be discussed in chapters 3 and 4.

This section illustrates the different architectural layers that can be used to design and develop the DT. In Section 2.4.7, we are going to clarifies the technologies currently available, and their contributions for designing and developing a DT.

2.4.7 Technologies related to DT

DT applications are now widely used in a variety of industries. DT solutions help diverse enterprises quickly identify areas for innovation and improve business processes and performance by providing precise virtual representations of real entities as well as simulations of operational processes. Product DT (e.g. the heart of the human body), operation DT (e.g. Magnetic Resonance Imaging (MRI) activity), and process DT (e.g. patient pathways), among others, are all examples of DTs that are used to visualize objects and processes, whether they are simple or complicated. According to Global Market Insight (Preeti Wadhwani, 2021), the Compound Annual Growth Rate (CAGR) of the DT market, which was valued at \$5 billion in 2020, is expected to increase 35% between 2021 and 2027.

A DT application incorporates five main technologies that aid in the creation of a digital representation, starting from the collection and storage of real-time data and the provision of important insights based on the information gathered as follows: (1) the IoT, (2) simulations, (3) Extended Reality (XR), (4) Cloud Computing, and (5) Artificial Intelligence (AI). A short summary of these technologies is illustrated in Figure 2.12.

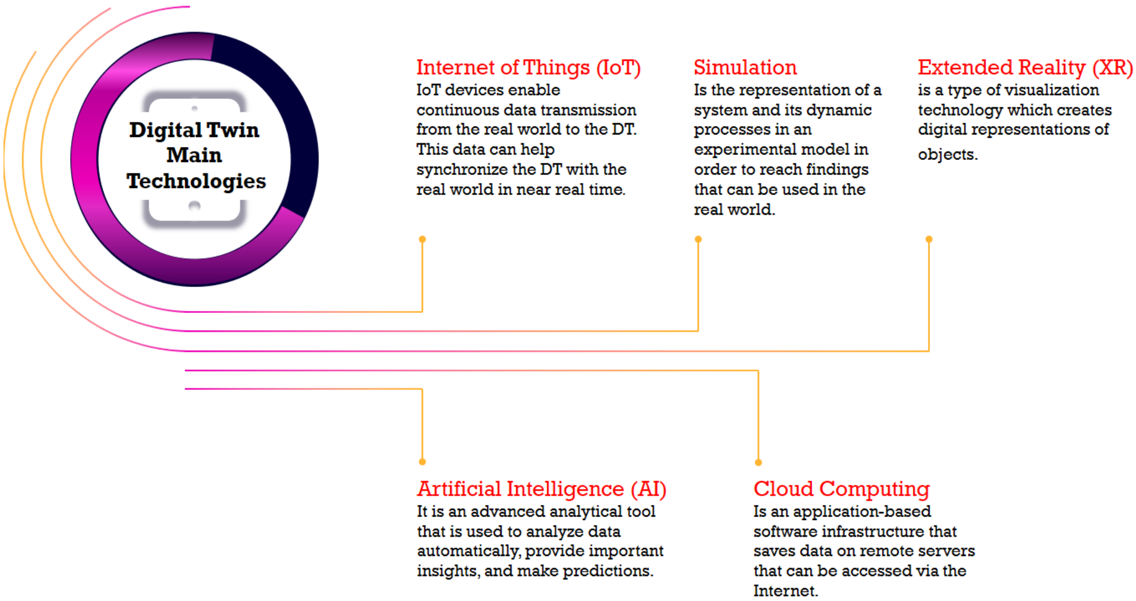


Figure 2.12: Technologies used in digital twins

1. **The Internet of Things**

   The IoT is defined as "a paradigm in which objects are equipped with sensors, actuators, and processors to communicate with each other to serve a meaningful purpose" (Sethi et al., 2017). In other words, the IoT is a network of interrelated objects, devices, and sensors that are able to interact with each other and exchange huge sets of data over the Internet. IoT can aid businesses improve efficiency, enhance health and safety, and improve the customer experience. According to McKinsey in (Manyika et al., 2015), the entire potential economic impact of IoT technology will grow from \$3.9 trillion to \$11.1 trillion per year by 2025.

   IoT devices are now part of our lives. They are already embedded in many businesses (industrial and healthcare). Reasonable use of these devices will help the real-time data collection and analysis that will contribute to better and more efficient data-driven decision-making. Most of the IoT wireless sensors are based on Radio Frequency

(RF) waves to transmit data to receivers: Wireless Fidelity (Wi-Fi), Near Field Communication (NFC), Bluetooth Low Energy (BLE), Ultra-Wide Band (UWB), Global System for Mobile (GSM) networks, and so on. These are examples of the technologies that can be used for IoT sensors. Two main functions for the sensors are Radio Frequency Identification (RFID) and/or Real-Time Location Systems RTLS. RFID and more recently, RTLS, are already used in many areas, as mentioned by (Halawa et al., 2020), from tracking in smart factories to counting items during warehouse inventory. Retail, manufacturing, logistics, smart warehousing and banking are among the major industries that use these IoT solutions. Moreover, according to a survey conducted by Gartner (Gartner, 2019), 75% of interviewed organizations working with IoT now employ DT applications.

It seems self-evident that "the rise of digital twins fits with the rise of the Internet of Things" (Hippold, 2019), because the vast amount of sensor data and device metadata generated by the IoT necessitates a system for organizing and managing all of this data, which is realized by the concept of the DT. Although the relationship between the DT and the IoT appears to be clear at first glance, IoT connects devices to the Internet and collects data, whereas the concept of the DT is used to represent, structure and manage that data so that it can be used for optimization and automation with the help of Artificial Intelligence and ML algorithms. (Jacoby et al., 2020).

In summary, IoT sensors linked to the real-world environment are used to collect data in real time on the state of this environment. This data will be transferred to the DT, which will reflect it in a virtual world that can be represented by a simulation tool. This will help the decision-maker keep track of the real-world situation, to anticipate its future, and perhaps improve its processes.

2. **Simulation**

According to (Shannon, 1975), simulation is "the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies (within the limits imposed by a criterion or set of criteria) for the operation of the system". DES, stochastic simulations, and deterministic simulations, among others, are all examples of simulations. However, the variables utilized in running these types of simulation are either known or random.

Simulation models are considered the core functionality of the DT. They can work together with AI and ML to analyze, understand, predict, and optimize complex systems (Siemens, 2020). According to (Negri et al., 2019), DTs could be developed by adding black-box modules to the main simulation model. This will aid in simulating different behaviors of the system (such as energy consumption, availability, kinematic behaviors, etc.). More details about the modeling, the simulation, and the relation between them exist in (B. P. Zeigler et al., 2018).

In short, "today's most complex products and processes are designed, tested and calibrated in the virtual world before being manufactured in the real world. Model calibration involves creating and simulating software models of future products, eventually leading to the creation of **digital twins**" (Siemens, 2021).

3. **Extended Reality**

With the invention of flight simulators in the early 1900s, simulation became the best practice for training (Bashir, 2010). Simulated training environments have advanced significantly during the last three decades, combining visual, audio, tactile, and motion components into a "virtual" reality (Higgins et al., 1997). Virtual reality (VR) is defined as "the interactions between an individual and computer-generated environment stimulating multiple sensory modalities, including visual, auditory or haptic experiences"

(Dascal et al., 2017). Real and virtual objects have recently been incorporated into training environments. Augmented reality, or (AR), is a simulation environment in which "the virtual world is superimposed on the real world such that both are experienced concurrently" (Vozenilek et al., 2004; Zweifach et al., 2019).

Extended reality (XR) is the visualization technology which creates digital representations of objects. XR capabilities enable a DT to digitally model physical objects, enabling users to interact with digital content. In fact, DT visualization might be a combination of virtual, augmented and mixed-reality (MR) simulations that simulate the real-world counterpart. The software digitalizes and simulates a real process, machine, or thing using real sensor data and models that allow users to interact in a virtual manner. Below is the description of AR, VR, and MR:

- Augmented reality overlays computer-generated content onto the real world and allows users to interact with it in real time (Alizadehsalehi et al., 2020). AR is the real-time integration of information such as text, graphics, audio, and other virtual additions with real-world objects.
- Virtual reality describes computer technologies that use software to generate realistic visuals, sounds, and other sensations that reflect an immersive environment and simulate a user's physical presence in it (Alizadehsalehi et al., 2020). In other words, VR is a computer-generated simulation in which a person can interact with an artificial, three-dimensional environment utilizing electronic devices such as special goggles with a screen or sensor-equipped gloves. The user can enjoy a realistic experience in this simulated artificial world. Everything is digitized and manipulable in virtual reality, allowing users to interact as if they were really there. VR uses headsets and goggles to give the illusion of a real-world environment, and it frequently incorporates sound and vibrations.
- Mixed reality is the merging of the real and virtual worlds to create new environments and visualizations in which physical and digital objects coexist and interact in real time (Basu et al., 2021).

4. **Cloud computing**

   Cloud computing is the delivery of computing services over the Internet (the cloud), including servers, storage, databases, networking, software, analytics and intelligence, in order to bring faster innovation, more flexible resources, and economies of scale (Sunyaev, 2020). Cloud computing does, in fact, play a vital part in DT applications. It is used to store and access data via the Internet more efficiently. Because DT applications deal with large volumes of data, cloud computing enables data to be stored on a virtual cloud and be accessed from anywhere.

   Using cloud computing for DTs will aid in the development and adaptation of the virtual entity entirely in the cloud, allowing developers, designers, and everyone else involved to gain access to the virtual entity and obtain the information they require to make significant decisions via the Internet, regardless of their location. As a result, it is critical to have linked devices with a high level of security to protect systems while also allowing connectivity and data transmission (Tao et al., 2019).

5. **Artificial Intelligence**

   This is a branch of computer science that tries to make machines think and act like humans. Machine vision, ML, domain knowledge, and other technologies are all part of this broad field. (Ma et al., 2021).

   AI is the digital replication of three human cognitive skills: learning, reasoning, and self-correction (Rathore et al., 2021). Digital learning is a set of rules implemented as a computer algorithm that transforms real-world historical data into useful information that can be executed. The purpose of digital reasoning is to select the best rules for achieving a particular goal. Digital self-correction, on the other hand, is the iterative process of accepting the results of learning and reasoning. To create a smart system, every AI model follows this process.

AI enables DT to be intelligent in the virtual world and make autonomous decisions based on data analysis (Autiosalo et al., 2019). ML is a type of AI. It is a collection of algorithms and models that can process large amounts of data generated by AI or a user in order to extract useful information and make appropriate decisions. Through the massive quantities of data analyzed, it allows systems to learn and enhance their performance and accuracy (Ahuett-Garza et al., 2018).

This section gave a brief overview about the common technologies that could be used to design and develop a DT. In fact, by going deeper into these categories, different techniques or methods could exist. If we take AI as an example, different techniques could include neural networks, fuzzy logic systems, genetic algorithms, particle swarm optimization, colony optimization, etc. Another example, regarding the IoT, means dealing with different devices, protocols and security aspects, among others.

The aim of this research work is to create a proof of concept of a DT for patient pathways in hospitals by using DES in combination with IoT technology. For example, real-time data acquisition is mandatory for transforming a traditional offline simulation model into a real-time online simulation model. For this reason, the IoT will be used to enable communication and data transmission between readers (or beacons) associated with each activity area to detect a patient, who must wear a special RFID or RTLS tag (embedded in a card or wristband). This RFID tag can be detected each time a patient is close to an RFID reader, for triggering a starting/ending event each time a patient goes through a door to go in/out of a room. In the case of an indoor geolocation function with RTLS, the location of the tag is updated at a given frequency (for example, at one-second intervals) using at least three beacons and a triangulation algorithm. A rule can be associated with the RTLS tag for triggering a start/end event each time it goes in/out of a virtual area corresponding to an activity. Whatever the solution, (RFID or RTLS), the collected starting and ending events will be used as input data for the DES model. Based on these events associated with each activity of the patient pathways, the state of the DES model can be updated in real time to be synchronized with the real world.

In this research, we did not focus on AI techniques or extended-reality approaches. For our experiments, we used a private cloud database that is installed on a server at IMT MINES ALBI. This database is used to store historical and current data. This data helps initialize and synchronize the DT with the real world. Before we go into our proposed approach for initialization and synchronization, which will be detailed in Chapter 4, Section 2.4.8 show the different approaches that used through the literature to illustrate the different mechanisms behind the initialization and synchronization the DT with real world.

2.4.8 Existing initialization and synchronization approaches

Based on the characteristics of the DT that have been discussed in Section 2.4.5, DT models should exist in a high-fidelity environment and reflect the reality of the real sensory entities in near real time in order to make meaningful decisions regarding the real sensory entities or to anticipate the effects of any unexpected events that may affect these entities. Two methods have been reported in the industrial sector literature in general and specifically under the domain of symbiotic simulation (online simulation) to achieve these characteristics: (1) initializing the simulation model (in our case, a DT) with the current state of the real world, and (2) synchronizing the simulation model to evolve alongside its real sensory entity. For example, in the case of any change to the physical entity, this change shall be mirrored in the simulation model (DT). Two methods have been described in the literature which seem to be relevant for achieving this purpose:(1) State Collection Approach (SCA) (Aydt et al., 2008), and (2) Base Simulation Approach (BSA) (Low et al., 2005).

The SCA is a periodic approach, and the initializing and synchronizing functionalities in this approach are totally dependent on the existence of a database that will be used as middleware, or a connection layer, that communicates data and information between the real world and the simulation model. The BSA is an event-driven approach. The initializing and synchronizing functionalities are dependent on the existence of an observer, or what is called a simulator (in our case, a DT for monitoring the real world: see Chapter 3).

In SCA, the data relating to the states of the physical system is periodically collected from the real world and stored in a database. This leads to synchronizing the database with the real world. In other words, at each time unit, the database will be fed the data on the physical system states from the real world. In this approach, the database is used as an observer for the real world. In the case of prediction, the most recent complete state of the physical system will be extracted from the database to be used to initialize the predictive simulation model. After that, the predictive model will be run faster than the real-world clock in order to predict the future, as illustrated in Figure 2.13. According to our view, the SCA seems to have a variety of weaknesses:

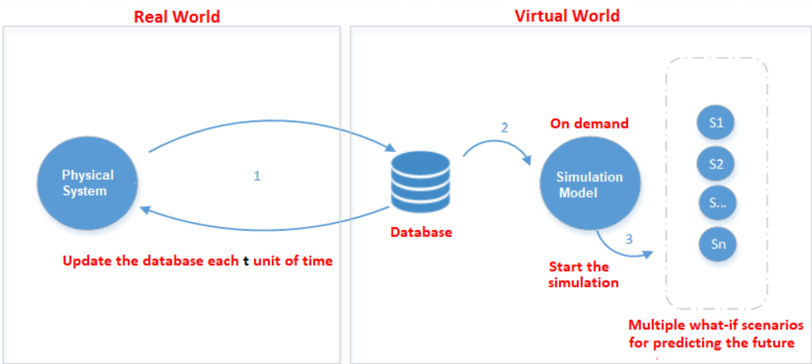


Figure 2.13: State collection approach

1. Overlapping problem: starting new updates on the database when the previous update has not finished. This problem can occur because this approach is a periodic approach; this means that at each constant time t, the state will be collected from the real world to feed the database.

   For example, assume that t=3 seconds: this means that the state of the real world will be captured every 3 seconds to feed the database. Assume that the gathering of real world states to feed the database begins at time=0. Assume that the time it takes to collect the states and reflect them in the database is 5 seconds (response time), although this time is not deterministic. In fact, this time is affected by the number of states that must be gathered from the real world. In this example, the second collection of states from the real world will begin at time=3 seconds, when the previous collection's response time will not yet have finished. In reality, there will only be 2 seconds left to complete the task.

2. Unexpected event problem: during the process of reflecting the states in the database, one or more unexpected events may occur. These events will not be reflected in the database until the second state collection begins.

3. Real-time problem (initialization problem): initializing the predictive simulation model with data that does not reflect the current situation of the physical system. Based on

this approach for predicting the future, the latest complete states from the database will be used as input for the predictive simulation model. In our opinion, the most recent complete states in the database may not match the current situation of the real world. In other words, some events might occur in the real world before the time for reflecting these events in the database has begun. As a result, using these states as input for the predictive model may provide inaccurate prediction results.

4. Technology dependent: this approach may be not suitable for the organization that uses discrete event technology, (e.g. RFID), because it collects the data periodically, which means some events will be lost. In other words, one or more events may have occurred in the real world during the database update. In this case, these events will not be taken into account.

In BSA, a simulator is used to observe (monitor) the physical system at each event detected in the real world. For example, in the case of an event that is detected in the real world, this event has to be injected into the simulator. In this case, the synchronization is done between the simulator and the real world with an event-driven approach. In the case of prediction, a clone of the simulator's current states will be loaded into the predictive simulation model, then the predictive model will be run faster than the real-world clock in order to predict the future, as depicted in Figure 2.14. The main issues that could exist in this approach are:

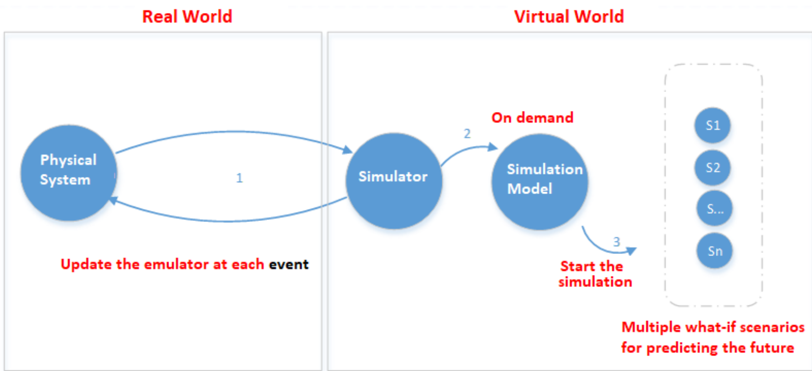


Figure 2.14: Base simulation approach

1. If there is an event that has not been injected into the simulator, it may cause the simulator to diverge from the current behavior of the physical system. In this case, cloning the states of the simulator into the predictive simulation to predict the future will lead to an inaccurate prediction.

2. Maintaining the simulator to be continuously running to observe the real world will consume the hardware and software resources used for the observing method. For example, continuously monitoring each patient at each point of time will consume more resources than monitoring him/her at each detected event. Furthermore, continuous monitoring means collecting a huge quantity of data, and it may require complex computation and processing techniques.

In comparing the SCA with the BSA, we observe that the two approaches reflect the current state of the physical system. The accuracy of this reflection differs in the two approaches. For example, the accuracy of the SCA depends on the value t, which represents the duration time between two updates of the database. For the BSA approach, the accuracy depends on how all of the events to be injected into the simulator are considered.

In our opinion, the BSA may be better than the SCA in terms of synchronization because this approach injects only the occurring real-world events into the simulator, rather than retrieving all of the states from the real world, as the SCA does. Second, synchronization occurs only at each detected event, as opposed to the periodic approach, which requires retrieving the state from the real world at each t unit of time regardless of whether there are any changes or not. However, in the industrial domain, there are different research works mentioning the two approaches, such as in (Cardin et al., 2011; Low et al., 2005) which mentions the BSA, or where (Katz et al., 1993) mentions the SCA.

In this research work, we are going to benefit from both approaches to develop a proof of concept for a DT that is able to monitor the patient pathways in the hospital in near real time and a prototype for a DT for prediction that is able to predict the near future of these pathways. In order to attain our goal, we will combine these two approaches. In other words, a real-time database will be used to monitor the patient pathways at each detected event which is not periodic. This will prevent any real world event from being missed, and it will help to initialize the DT with the current state of the hospital. Moreover, this database will also be utilized as a backup for historical events, thanks to its existence. It will aid in analyzing the hospital's behavior and locating bottlenecks by replaying earlier events. In addition, in case of failure or divergence, this database will be used to re-initialize the DT as needed. Furthermore, this database will be able to run not only the previous event, but also a certain period in the past. In addition, all of the saved events will contribute to the creation of an updated dynamic distribution that can be used to anticipate the near future. This historical data will be used to run a prospective analysis, and the current data will be used to run a predictive analysis. Likewise, a 3D dynamic virtual representation for these pathways will be developed, which is called a DT for monitoring. This DT will work as an observer (monitor) for the patient pathways. This will allow us to have a 3D view of the current state of the hospital. More details about our proposed approach are available in chapters 3 and 4.

2.5 Conclusion

The purpose of this research review is to help the reader understand the different aspects depicted by the research on the subjects related to our objective, which is to use a DES to design a proof of concept for a DT for patient pathways in the hospital, named HospiT'Win. This DT is able to monitor in real time the patient pathways in the hospital and to predict their near future. Different subjects have been discussed, starting from understanding the patient pathways in the hospital, then following with DES and finishing with the DT.

In the literature there is no specific definition for patient pathways in hospitals. For this reason, we have proposed our own definition, after analysis of several proposed definitions. The techniques that have been used to improve patient pathways are illustrated in this chapter, in addition to their advantages and weaknesses. Furthermore, a DES definition and its main components have been clarified in detail. The strengths, the weaknesses and the application domain in health care related to DES have been explained.

For the DT, four different definitions were found, but as there was no common definition, we classified the identified definitions by application domain. However, there was no definition that fit this research work, which is a DT for patient pathways. For this reason, we have proposed our own definition. Different issues relating to the methodologies of developing the DT and initializing and synchronizing the DT with the real world have been illustrated.

In conclusion, there are different application domains where the DT has been applied. Most of the works have been done in the industrial domain. Fewer works have been done in the healthcare domain, and little attention has been paid to hospital management, which is the core of our research. Moreover, different types of DTs have been developed, such as product, operation, and process DTs. These types have been used in different ways, starting from learning and designing, and ending with preventing. This work will use a DT, and the main purpose of the DT will be for monitoring and predicting patient pathways in the hospital.

Most of the tools that were discussed in the literature are based on offline approaches using historical data. In other words, these approaches use data that can be extracted in a non-significant period of the past. This means that they do not have any up-to-date data because they are not connected to the real world in real time. In this research work, we will assess and benefit from this online concept called the DT to improve patient pathways in the hospital.

After this state of the art, we can say that there are no standard or comprehensive methodologies able to help in designing a DT for patient pathways in the hospital. Thus, some scientific research questions have been developed to design this kind of this DT, as illustrated in Figure 2.15.

Figure 2.15: Scientific research questions

1. How can DES be used to design a decision support tool based on a digital twin for real-time monitoring of patient pathways in the hospital and predicting their near future?

   Answer: in Chapter 3

2. How can the digital twin have an accurate vision of the current situation of the patient pathways?

   Answer: in Chapter 4

3. How can the Digital Twin for Monitoring (DTM) be triggered to run the Digital Twin for Predicting (DTP)?

   Answer: in Chapters 4 and 5

3

Methodology for Building HospiT'Win

3.1 Introduction

As stated in the previous chapter, a digital twin (DT) is an emerging technology that offers interesting opportunities for all organizations. It allows them to improve efficiency and operational productivity and reduce risks by using online simulation, real data analysis, and Artificial Intelligence (AI) capabilities. Accordingly, and given our social issue relating to ensuring efficient hospital management, this research attempts to design a DT dedicated to monitoring and predicting the near future of patient pathways in hospitals.

To achieve these objectives, we would need to adopt a methodology for designing a DT that properly formalizes the system's implementation requirements. However, as we discovered during our scoping of the health DT literature, there is very little information available regarding the steps to take and their associated engineering methods for developing a virtual representation of the patient pathway and connecting it to the real world.

The aim of this chapter is to describe our proposed methodology for developing a DT for patient pathways inside the hospital of the future, which is the hospital that moves from traditionally based systems to information technology based systems, using the Internet of Things (IoT), the Body Sensor Network (BSN), modeling, simulation, and AI (Karakra et al., 2019). However, the developed DT will be used as a decision support tool for monitoring the patient pathways in near real time and for predicting their near future with the goal of improving hospital management agility and efficiency, as well as patient satisfaction. Hence, several achievements have been made, including:

1. A real patient pathway meta-model that highlights the different elements in the real pathway with an organizational focus as well as the relationship between these elements.

In fact, understanding the real patient pathways allows them to be digitized more easily.

2. A patient pathway DT meta-model that captures the required elements and the relationship between these elements in order to design the proposed DT. The primary goals of this meta-model are to highlight and capitalize on the required elements from the Discrete Event Simulation (DES) tool, as well as on their counterparts in the real world, and the elements required to make the connection with the real world.

3. Dedicated process flow modeling elements which capitalize on and formalize a set of knowledge about the patient pathways as an initial step before developing the proposed DT. These process flow elements will allow a rich model to be generated that is able to distinguish multiple types of knowledge. Some of these types knowledge will be applied to offline simulation, while others will be applied to the DT. The motivation for this contribution stems from the reality that there is currently no universal technique for modeling patient pathways with a rich level of knowledge, and each institution must specify its own format and level of detail.

These achievements have been made in accordance with the definitions and the assumptions listed in Section 3.2.

3.2 Definitions and Assumptions

The scope of this research work is the management of patient pathways in hospitals, which represent ecosystems made up of living and non-living components with close relationships. The living components represent the hospital population and the non-living components represent equipment; for instance, the patient is considered a living component and the registration desk is considered a non-living component.

> In this setting, the patient pathway in the hospital can represent the connection between the aforementioned components, and in this work it is defined as a sequence of activities (operations/actions) that a patient undertakes or does inside the hospital. Four types of these activities are considered: clinical activities, non-clinical but required activities (administrative activities), waiting activities, and movement activities (moving between two activities).

> Accordingly, the DT of patient pathways in hospitals can be defined as a high-fidelity and dynamic virtual representation of the patient pathways in the hospital, created with data that is historically available. It is also designed to collect real-time data continuously from the real patient pathways. This DT is capable of providing information about the past states, the current states and the future states of patients.

To design, validate, and implement this DT, both the historical and up-to-date sensory data from the real world hospital are required, in addition to the real hospital itself. To have these requirements from this critical facility, some issues must be resolved first. For example, (1) the patient data has to be kept confidential, (2) sensors must be set up inside the hospital for collecting real-time data, (3) the population (patients, staff) of the hospital must agree to being monitored, (4) the hospital activities must not be affected while running the DT, and (5) the decision-maker and the hospital manager must trust the DT. Given the difficulty of implementing these measures in a real-world environmental context while developing our methodology, some assumptions were developed as a first step:

A1: *We assume that the patient pathways in the hospital can be depicted as a continuous flow representing a collection of states that is sufficient for describing the execution of patient activities at any given time. These states change with the occurrence of an event.*

Example: the state of the patient who walks from the hospital entrance to the registration desk is **walking**. When the patient arrives at the registration desk, the state of the patient changes from **walking** to **waiting** for administrative processing. This change of the patient's state happens due to the occurrence of the event called **arriving** at the registration desk.

A2: *In this work, we assume that there is a hospital information system that has recorded past events about the patients; at a minimum, the patient identification and the time when the patient arrives or leaves the activity and the hospital, in addition to the duration of time the patient has spent at each activity.*

> To remove any ambiguity, it is worth mentioning that in our meta-model, we have an activity called **moving activity**. This activity represents the patient when he/she moves from one place to another. This activity could be described by a set of states. In this work we used **walking state**. This means that when the patient is moving, his/her state is **walking**.

A3: *In this work, we assume that a real hospital has a patient tracking system that collects data on the patient when he or she enters or exits an activity or the hospital. This information will be used to notify the DT of the patient's location. This will assist the DT in reflecting the patient's location within the model.*

Example: when a patient arrives at the hospital entrance, the sensor attached to the door will detect him or her. In this case, the sensor will notify the DT that a real patient has arrived at the hospital. In this case, the DT will create a virtual patient (virtual twin) representing the real one at the entrance of the virtual hospital.

A4: *In this work, we assume that all the resources (human, technical, fixed) have an infinite capacity, and all of the required resources to start the activity are available at the time of starting the activity.*

Example : we assume that when the patient arrives at the exam room activity, all of the resources required to perform the activity are available, including human resources (a doctor, nurses), fixed resources (beds), and technical resources (equipment needed for examining the patient).

3.3 Proposed Methodology

After reviewing the available scientific literature about the usage of the DT in the healthcare domain, we have found that little attention has been paid to hospital management. To be more precise, the DT that is used in hospital management is limited to a single unit, department, or activity such as the Intensive Care Unit (ICU) or the Emergency Department (ED). To the best of our knowledge, no research work has been done to study the whole patient pathway in the hospital based on a DT.

By looking at the hospital as a complex ecosystem, a network of activities that are interdependent can be found. For example, after the registration activity, the patient may go to the waiting room, and after the waiting room, the patient may go to the exam room. Thus, from this small example we find three activities that depend on each other. As a result, improving one activity may have a positive or negative impact on the other activities. Improving the registration desk's performance to serve more patients when there isn't enough space in the waiting room may lead to a bottleneck in the area after registration and before the waiting room. On the contrary, if the waiting room is large enough to accommodate more patients, improving the registration desk's performance may provide a benefit to the whole system. Instead of focusing on a single activity, we propose a methodology for designing a DT that can monitor the whole patient pathway to detect any variations or bottlenecks or any service dependency in real time.

This section outlines the methodology followed in this thesis for designing a digital twin of patient pathways in a hospital. Figure 3.1 depicts a general overview of the proposed methodology for developing a DT of patient pathways. This methodology has three phases and four steps. The phases of this methodology are the design phase, the ramp-up phase and the running phase, while the steps of this methodology are construction, validation, transformation, and deployment.

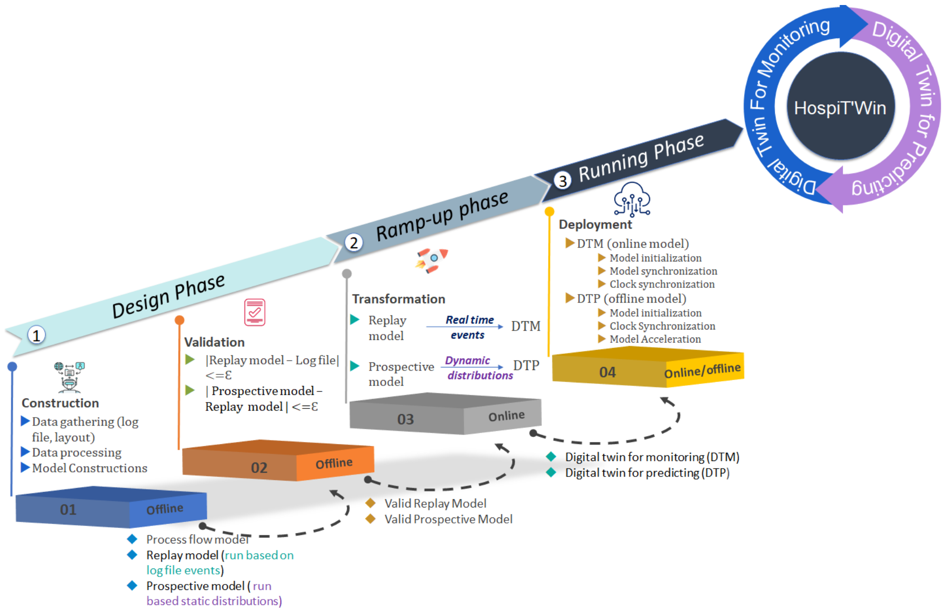


Figure 3.1: Proposed methodology for designing and building HospiT'Win

The first two steps (construction and validation) are done offline without connecting with the real world, and the data used are historical. The second step (transformation) is done online and is connected with the real world, and the data used are real-time data, whereas the third step (deployment) is done online for running the Digital Twin for Monitoring (DTM) and (semi-online)[1] for running the Digital Twin for Predicting (DTP), but both DTs are initialized with current data. This methodology begins with the collection of two types of data: (1) the hospital's architectural layout, and (2) historical data from the Hospital Information System (HIS) in the form of a log file that represents past events that have occurred in the hospital. The data gathered will be used to build three knowledge models: a process flow model and two offline simulation models known as a replay model and a prospective model. Other sections will go over the specifics of the proposed methodology depicted in Figure 3.1.

[1]The DTP is connected with the real world only to have up-to-date data (real world current state and updated distributions). After that, it continues working offline for predicting the future. In this work, we consider this type of DT an offline DT.

3.4 HospiT'Win Phases and Steps

3.4.1 Design phase

This phase is considered an offline phase, which means this phase is not connected with the real world in either step, whether it is the model construction step or the validation step. In this phase, we are aiming to answer the following questions:

- What are the minimum requirements needed to start building a DT of patient pathways in a hospital?
- What are the different types of application design packages that could be used to design a DT model?
- What are the different types of models to construct before the DT is built?

To answer the questions above, it should first be noted that there are different types of DTs with different usages and requirements. For example, the requirements for DTs for monitoring differ from the requirements for DTs for prediction, and they also differ from the requirements for DTs for optimization, although this third category does not fall within the scope of this research. For example, a DTM needs to be synchronized with the real world in real time at each detected event, while the DTP and the Digital Twin for Optimization (DTO) do not. Another example: the DTP uses the current state of the hospital as an input to predict the future, whereas this is not the case with the DTO. For us, the output statistics of the DTP could be used as an input for the DTO to optimize the patient pathways for a specific target. To achieve the optimization procedure, an optimization strategy is required (for example, an objective function, some parameters, optimization algorithms, and so on). In this research, we have set up a methodology for designing a proof of concept for a DT for monitoring patient pathways and developing a prototype for a DT for predicting their near future.

Before beginning to design the proposed DT models, the following requirements must be met:

- Hospital architectural layouts can assist in designing the virtual hospital's building structure, determining the number of floors, rooms, and locating the various architectural items and the walk path in a manner that corresponds to the actual hospital.
- Historical data can aid in locating the various activities that occur in the hospital, the types of these activities, the resources needed to carry out these activities, the length of time that these activities take from start to finish, and the number and the identifications (IDs) of the patients who have visited these activities, in addition to other information.

  To achieve the goal of this research in using the DES tool, the historical data can be transformed into a log file of events that includes the following structure: {PatientID, Timestamp, EventDescription}. The **PatientID** in this structure corresponds to the identification of the patient who generated the event, and the **Timestamp** corresponds to the time of the occurrence of the event, whereas the **EventDescription** describes the type of event that occurred, such as a starting event or an ending event. An example to illustrate this structure: suppose the patient with ID 3456 arrives at the entrance of the hospital at 7:00:00 am on 21/06/2020, and he walks forward to the registration desk through the waiting line. The patient arrives at the waiting line at 7:05:30 am and he waits three minutes. The patient leaves the waiting line at 7:08:30 am, and he arrives at the registration desk at 7:08:50 am, and so on. From this example, we have four events that describe part of the patient pathways as the following: (1) {3456, 7:00:00 21/06/2020, arrive hospital entrance}, (2) {3456, 7:05:30 21/06/2020, arrive waiting line}, (3) {3456, 7:08:30 21/06/2020, leave waiting line}, (4) {3456, 7:08:50 21/06/2020, arrive registration desk}. Thanks to this structure, we will be able to discover the different patient pathways in the hospital by using a process mining tool such as Disco (Fluxicon, 2021).

Different universal application design packages can be used with the above knowledge to design the DT, such as Computer-Aided Design (CAD), Computer-Aided Engineering (CAE), and Computer-Aided Manufacturing (CAM). Some of these packages, such as CAD, including AutoCAD, CATIA, Fusion 360, NX, and SolidWorks among others, can only generate diagrams or 3D models without the ability to simulate the dynamic behaviors of the processes. On the contrary, CAE applications such as Abaqus, Ansys, CFX, Comsol, Excel, Fluent, HEEDS, HyperWorks, LS-DYNA, Matlab, Nastran, Simulink, and STAR-CCM+, among others, are used to simulate the effects of various conditions on a product's architecture. On the other hand, CAM is used to automate manufacturing processes. Two or more packages must be combined together to provide a powerful, full set of features/functionalities such as design, track/monitor, simulate, analyze, control, evaluate and optimize at the same time. However, the application design packages mentioned here are out of the scope of this research.

This work aims to benefit from discrete event simulation packages such as FlexSim (FlexSim, 2021) with the help of the IoT to design the DT for patient pathways, as it is easy to use these packages to design and to develop a live DT which is able to simulate, to anticipate, and possibly also to control and/or manage the behavior of the real world.

Before designing the proposed DT models, we have developed two meta-models: one meta-model corresponding to the real patient pathways in the hospital, as will be discussed in Section 3.4.1.1, and another meta-model corresponding to the DT of patient pathways in the hospital, as will be clarified in Section 3.4.1.2. We have also developed a process flow modeling notation that conforms to a developed DT meta-model. These elements will be described in Section 3.4.1.3. These meta-models, in addition to this process flow modeling notation, will assist the designer and the developer in designing and developing a DT of patient pathways in the hospital.

3.4.1.1 Proposed meta–model of real patient pathways in the hospital

The first step towards designing a DT for patient pathways in hospitals is to understand the main elements that form the real patient pathways and the relations between these elements. In fact, the DT is a virtual representation of the real sensory entity. Understanding the real entity allows it to be easily digitized. Thus, we have developed a meta-model for the real patient pathway that highlights the common elements that will be used in this research work, as illustrated in Figure 3.2.

In this meta-model, the hospital is a composite of patient pathways and resources, and it also aggregated from the structural building. The hospital has a communication network. This network is a composite of the main elements that provide the connections between the real patient pathways with a DT of patient pathways such as servers, IoT sensors, tags, etc.

Patient pathways as the core component of this meta-model are aggregated from activities and decisions. The decisions are responsible for determining the next activity the patient may follow (e.g., after registration the patient may go to the waiting room, etc.). Each activity has a duration time; this represents the length of time that the patient spent inside the activity. Three different types of activities can be found in the real world: (1) Value Added (VA) activity which represents any activity that adds a value to the patient care (e.g., clinical activity), (2) Non-Value Added (N-VA) activity, which is any activity that may add cost or time without adding any value to the patient care (e.g., waiting activity,

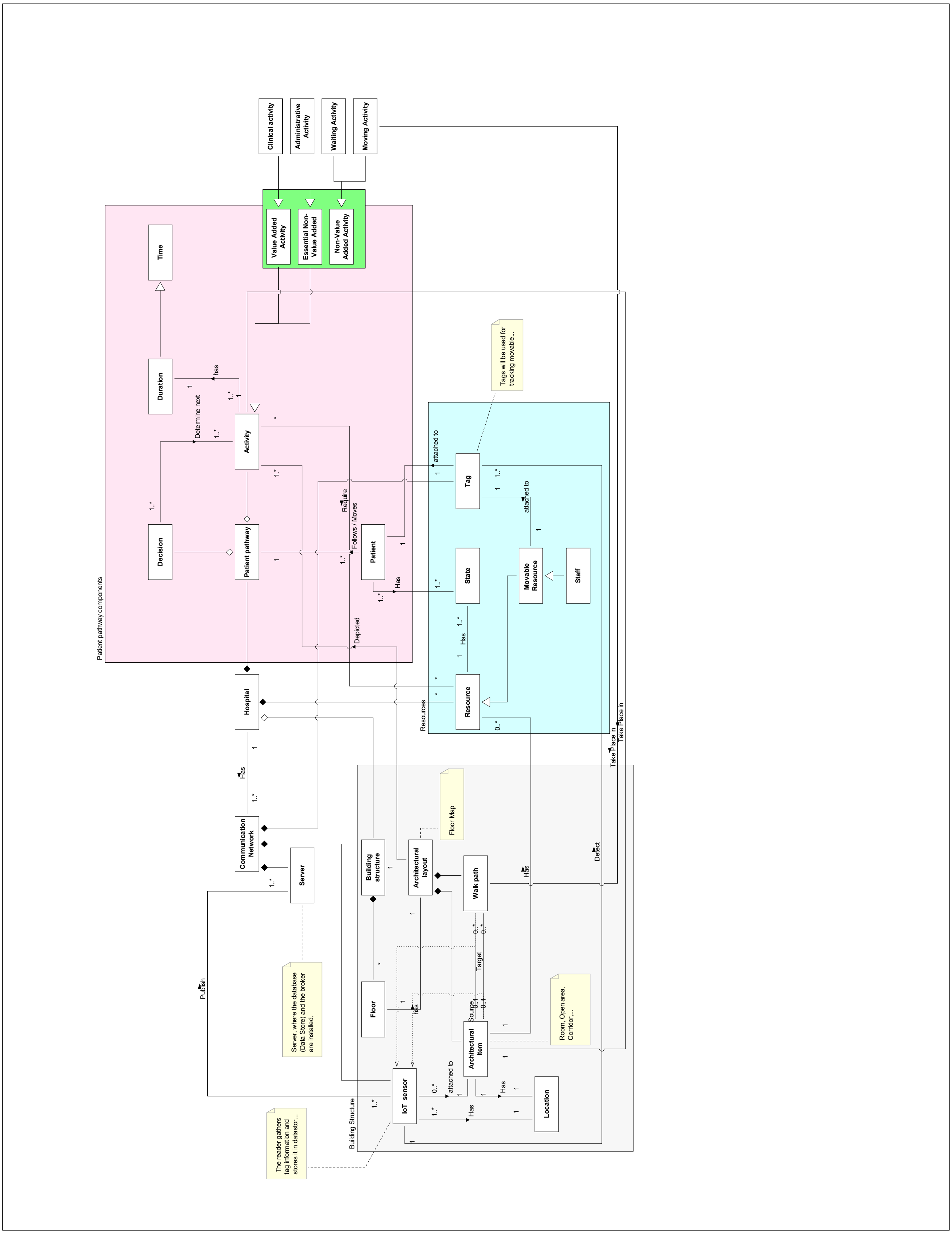


Figure 3.2: Meta-model of patient pathways in a hospital

moving activity), and (3) Essential Non-Value Added (ENVA) activity, which represents any activity required in order to start another activity, but adds no value to the patient (e.g., administrative activity).

The second component is the structural building of the hospital. In fact, the building's structure is a composite of the floors, where each has an architectural layout (map), and each architectural layout is a composite of walk paths and architectural items (e.g., a room, open area, etc.). A walk path is used to connect two architectural items together. Normally, each architectural item is attached to an IoT sensor to detect the patient when he/she enters or leaves this architectural item. As soon as the patient is detected by the IoT sensor, thanks to the tag worn by the patient, event information about this patient will be published to different servers in the communication network. Moreover, each architectural item has a location (e.g., floor number 1, room number 2, etc.). Each activity takes place inside an architectural item, except for the moving activity which takes place on a walk path.

The third component of this meta-model is the resources. Different types of resources can exist (e.g., movable resources). Each resource has a state, and these resources also have a place inside the architectural items. Moreover, the activity to be executed requires different types of resources.

In this meta-model, the patient wears a tag, and when the patient enters/leaves an architectural item, this patient is detected by the IoT sensors and this event information about this patient is published in the communication network to inform the DT that an event has occurred in the real world. However, another meta–model that captures the different elements needed to build a DT for patient pathways has been developed in Section 3.4.1.2. The relation between these two meta-models will be summarized in Table 3.1.

3.4.1.2 Proposed meta–model of a DT of patient pathways in the hospital

This section aims to answer the following questions:

- What are the various elements that must be captured in order to design a DTM and a prototype of a DTP of patient pathways in the hospital?
- What is the relation between the elements that have been captured?
- Are there any differences between the DT elements and the real patient pathway elements, and what is the relation between them?

From our standpoint, running an online DES model connected in real time to the real world corresponds to having a DT. Hence, understanding the patient pathways from the point of view of an online DES is an essential step because we are going to create a dynamic virtual environment using a DES tool that will be a decision support tool based on a DT for the real environment. This DT is a composite of two models: a DT for Monitoring (DTM) and a DT for Predicting (DTP). The DTM must be able to mimic the real-world environment and must be able to receive the data and communicate knowledge from/to the real world, whereas the DTP must be able to predict the near future of this environment. Thus, the mapping between the virtual world and the real world must exist in order to understand the relation between the two worlds. As a result, communication between the two worlds will be easy to create.

Figure 3.3 depicts a DT meta-model of patient pathways. This meta-model captures all the required elements that are needed for developing a DTM and a prototype of a DTP of patient pathways and the relation between these elements according to the DES and patient

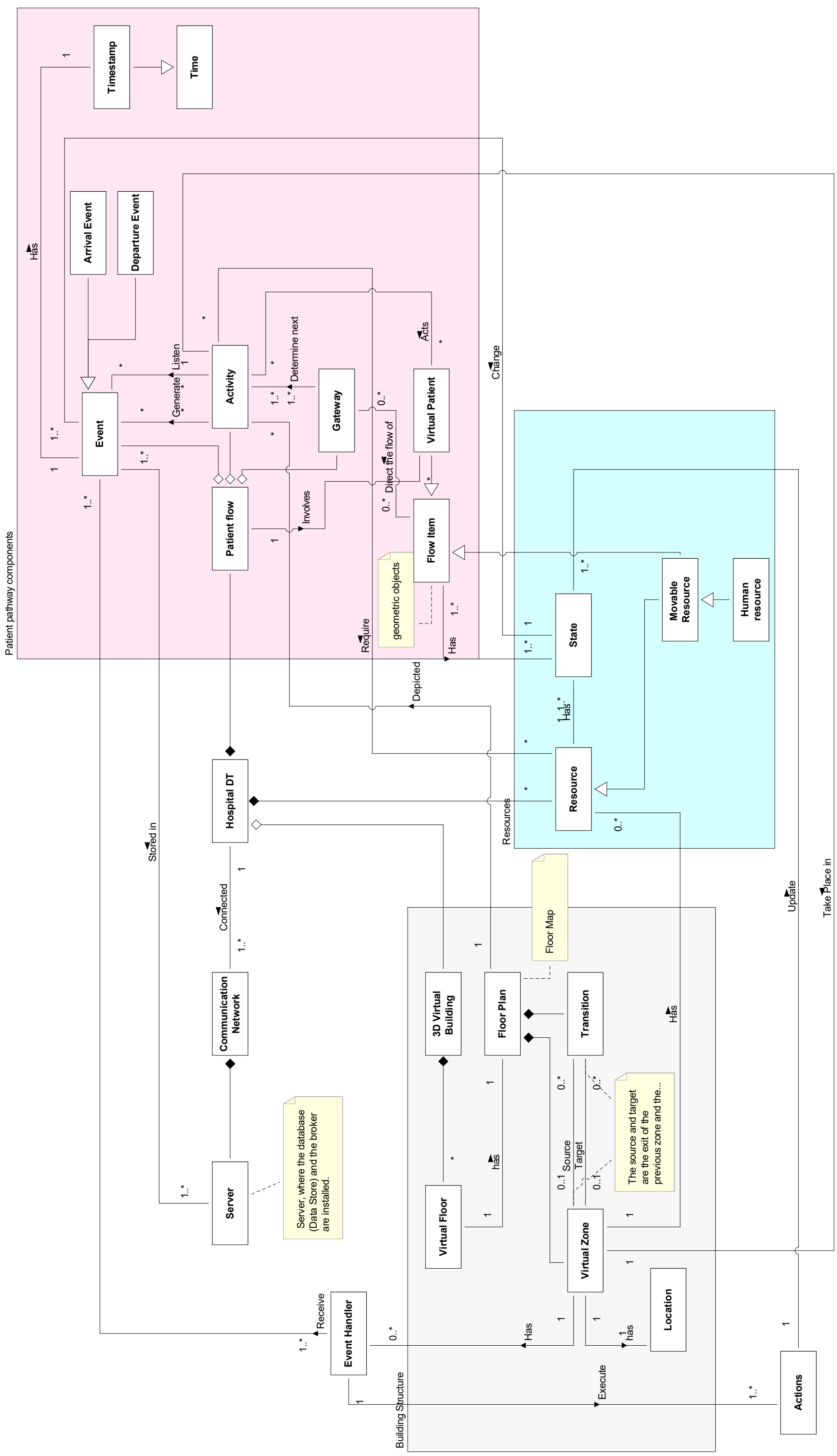


Figure 3.3: DT meta-model of patient pathways in a hospital

pathways in the real world. This meta-model has been developed as a formal architecture to aid the designer and the developer in generating such a proposed DT.

The hospital DT is composite of patient flows and resources, where patient flow represents a patient pathway in the real world. Moreover, this DT is an aggregation of a 3D virtual building which corresponds to the building infrastructure in the real world. This meta-model describes the connection between the hospital and the communication network, where the communication network is a composite of various servers that enables communication between the real hospital in the real world and the DT in the virtual world. These servers can include data storage functions for storing data and information transferred from/to the real world and the DT, in addition to services that allow wireless sensor networks to be set up using technologies such as Radio Frequency Identification (RFID), or the Indoor Real-Time Location System (IRTLS), among others.

Patient flow is an aggregation of events, activities and gateways (decision points in the real world). In the DTM, the activity is considered to have begun if it receives an arrival event from the real world, and it is considered to have ended when it receives another event called departure event, as shown in the listening relation between the activity and the event. Each of these events has a timestamp which is considered time. The gateway is used to determine the next activity the patient may perform (e.g., the activity after registration could be a waiting room or an exam room). Also, the gateway is used to direct the flow of the patient, using the different rules and stochastic probabilities (e.g., 30% of patients will go to the exits, while 70% of patients will go to the waiting room).The **Acts** relation that appears between the activity and the virtual patient is used to clarify that the activity may perform a specific action on the patient (e.g., the registration activity will register the patient, the treatment activity will treat the patient, and so on).

The 3D virtual building is composed of different floors. Each floor has a digital floor map (floor plan) which represents the architectural floor layout in the real world. Each floor plan is composed of virtual zones and transitions. The transition is used to connect two virtual zones. Each virtual zone has a location (e.g., floor 1, room 1). In addition, each virtual zone has an event handler; when the event handler receives an event from the real world, the event handler updates the state of the patient (accelerate patient, remove patient, etc.), as will be discussed in Section 4.4.2. In reality, each activity takes place inside the virtual zone except the moving activity which takes place inside the transition. To execute the activity, different types of resources can be used.

Table 3.1 illustrates the mapping between the different elements identified in the real world patient pathways with corresponding elements in the DT of patient pathways. It is obvious that different elements can exist in the real world where there is no synonym for them in the DT and vice versa. For example, in the real world, patients can wear a tag to be tracked inside the hospital, whereas this tag is not represented in the virtual world. In the real world, there are different types of activity, whereas from the point of view of the DES tools, all of these activities are considered to be one type. For example, VA activity, ENVA activity, and N-VA activity are considered different types of activity for the real world, but for the DES tool, all of them are considered one type, called activity, and the responsibility of the DES designers and/or developers is to distinguish among these activities by providing them with attributes such as names, descriptions, identification numbers (IDs), and so on.

As shown in Table 3.1 there are many elements in the real world that have their own synonyms in the DT. For example, tag information in the real world is a synonym for the event in the virtual world, even if the tag has less information than the event, but both of them represent information about the patient. Also, a patient in the real world is considered a flow item in the DT, even though the flow item can represent other types, such as movable equipment and human resources in the DT.

This section describes the different components and elements of the DT of patient pathways, and includes DES elements in addition to elements related to the way of connecting the DT

#	Patient pathways in the real world	DT of patient pathways
1	Hospital	Hospital DT
2	Patient pathway	Patient flow
3	Decision	Gateway
4	Activity	Activity
5	Tag	...
6	Tag information	Event
7	Patient	Virtual Patient, which is considered a type of Flow Item
8	Duration	= Timestamp of Departure event - Timestamp of Arrival event
9	Time	Time
10	Building Structure	3D Virtual Building
11	Floor	Virtual Floor
12	Architectural layout	Floor Plan
13	Walk path	Transition
14	Architectural Item	Virtual Zone
15	IoT sensor	...
16	Resource	Resource
17	State	State
18	Movable Resource	Movable Resource, which is considered a type of Flow Item
19	Staff	Human Resource, which is considered a type of Movable Resource
20	Timestamp	Timestamp
21	Location	Location
22	...	Actions
23	VA Activity	Activity
24	ENVA Activity	
25	N-VA Activity	
26	Has a Communication Network	Connected to the Communication Network

Table 3.1: Mapping between the real world patient pathway elements and the DT of patient pathway elements

of the patient pathway with the real patient pathway. According to this meta-model, some modeling notation constructs are proposed in Section 3.4.1.3 to develop a rich process flow that has as much information as possible about the real patient pathway. This will aid the designer and the developer of the DT to transform this process flow into a DT of patient pathways with less time and effort.

3.4.1.3 Proposed process flow modeling notation

A blueprint of the separate steps of patient flows in sequential order must be illustrated in a way to make the transformation from the conceptual model to the DT model easy. This picture can be illustrated in a rich process flow model. This model will be used to capitalize on and formulate some knowledge, such as the different pathways the patient may follow, the various activities the patient may be involved in, different gateways that direct the flow of patients, stochastic distributions, statistical probabilities, and the type of synchronization at each activity the patient does or undertakes. All of this knowledge is required for helping in designing a DT model in an easier and faster way. For this reason, we sought to find a process modeling language to meet these requirements. As a result, we compared three well-known standards according to our needs: (1) the Japanese industrial standard, (2) Business Process Modeling and Notation (BPMN), and (3) Extended Event-Driven Process Chain (eEPC), as shown in Table 3.2, in order to determine which one met our requirements best. However, we found that no single one of them was able to fit all of our needs. We opted to extend the Japanese industrial standard process chart elements (JIS, 1982). Due to the reduced time and effort required for us, we found that the Japanese industrial standard was the easiest to extend.

However, achieving the aforementioned requirements by adding new modeling notations will reduce the design time of the DT, and it will provide more knowledge for the DT designer to

Engineering Point of View		Modeling Language Notation Elements		
Feature	**Example**	**Process chart "Japanese industrial standard"**	**BPMN "OMG standard"**	**eEPC "ARIS"**
How to distinguish the different types of activities	Administrative activity: registration activity clinical activity: operation activity, etc.	-	-	-
How to connect the different elements in the graphical model	Connect the entrance with the waiting line (walk path).	+ <Flow line>	+ <Sequence flow symbol >	+ <Control flow symbols and/or info>
How to represent the different patient pathways	Patient A follows pathway 1, Patient B follows pathway 2.	-	+ < Gateway>	+ <Rules>
How to depict stochastic rules	About 30% of patients follow pathway 1, while 70% follow pathway 2.	-	-	-
How to represent the event when the patient arrives/leaves the hospital/activity	The patient arrives at the hospital entrance.	-	+ <start and end event symbols >	+ <event symbol >
How to represent that the activity is connected with the real world	When the real patient arrives at the registration activity, the corresponding virtual activity will be started.	-	-	-
How to represent whether this waiting (delay) requires or does not require resources	The waiting room requires chairs, while the waiting line does not require chairs.	-	+/- <There is a timer for delay but does not mention with/without resource>	+/- <There is no representation for the delay, but there is a representation for a resource>
How to represent the moving activity	Patient moves from activity A to activity B (transportation)	+ <transportation>	-	-
Total number of requirements can be achieved by each modeling language		2 requirements	3 requirements	3 requirements
+: fully supported, - : Not supported , +/- Yes; there is no notation, but in some way it could be represented.				

Table 3.2: Process flow modeling requirement notations

consider during the design phase. Moreover, these requirements will help in differentiating the different type of activities such as the VA, N-VA, and ENVA. In fact, all of these activities for the simulation tool are considered one type of activity, whereas in reality, these are different types.

Table 3.3 illustrates the modeling elements (notations) that will be used in our research work to draw an enriched process flow model that is needed to be transformed into a DT model. These modeling elements are designed to conform with the meta-model developed in Section 3.4.1.2. The relation between the designed elements and the meta-model are illustrated in Table 3.4.

Table 3.3 distinguishes different types of events that could be used to represent the synchronization between the virtual world and the real world. Symbol 9 represents a discrete event from the continuing connection with the real world, which means the activity is continuously executed. The start and the end of this activity are determined by two events: when the sensor detects a patient arriving at the activity, and when the sensor detects the patient leaving the activity. This is the reason behind the name "discrete events from a continuing connection". When this symbol is used in the process flow, it means the events will be gathered at the beginning and at the end of the activity.

Symbol 10 represents a continuing connection with the real world. When this symbol is used, it means the patient will be tracked at each point ing time, such as when using an RTLS. This symbol is outside the scope of this work because the main focus of this research is discrete events.

Symbol 11 illustrates an instantaneous connection with the real world. This symbol is used to represent just one event: for example, synchronizing the virtual patient with the real patient at the time the real patient arrives at the hospital entrance or the hospital exit.

#	Symbol	Description	#	Symbol	Description
1		In	8		Transportation (Move)
2		Out	9		A discrete event from the continuing connection with the real world (continuous activity)
3		Delay with occupying resources (chairs)	10		Continuous connection with the real world (continuous activity)
4		Delay without occupying resources (waiting line)	11		Instantaneous connection with the real world
5		Medical Operation	12		Connect elements
6		Administrative Operation	13	Label (Description)	Description for the symbol (number, percentage, distribution, text)
7		Gateway (AND, OR, XOR)	14	label	Group with label (ward, floor, etc.)

Table 3.3: Process flow of patient pathway modeling elements

Table 3.4 illustrates the relation between the meta-model developed in Section 3.4.1.2 and the modeling elements detailed in Table 3.3. For example, **In** and **Out** elements of the process flow will be used to represent the entrance and the exit of the hospital in the virtual world, and these elements can correspond to events in the developed meta-model, as well as the **Delay with or without occupying resources** in the process flow that corresponds to an activity in the developed meta-model. Furthermore, **Transition** in the meta-model is represented by a move element in the process flow and so on. The **Connection** elements in the process flow are represented by more than one meta-model element. In fact, the relation between some meta-model components illustrates these modeling notations. For example, when an event handler receives an event from the real world, the event handler will execute actions corresponding to the received events. This relation could represent any type of connection illustrated in Table 3.4 and depends on the technology used (continuous monitoring or discrete event monitoring technologies).

The following use case illustrates the process flow for a simple patient pathway in order to clarify the use of the designed modeling elements. In this pathway, there are three activities: Waiting Line (WL), Registration Desk (RD), and an Exam Room (ER). There is one sensor[2] at each activity that detects the real patient when he/she starts/finishes the activity, and there is a sensor that detects the patient when he/she enters or leaves the hospital. First, when the real patient arrives at the hospital entrance, the sensor associated with the door will recognize this patient, and the event information about this patient will be submitted to the Digital Twin for Monitoring (DTM)[3] to create a virtual patient at the virtual entrance door. At the same time, the real patient will continually move until he or she arrives at the waiting line. The patient will be detected by a sensor attached to the waiting line, and the event information will be sent to the DTM to reflect it. The sensor detects an event when the patient leaves the waiting line. In order to release the virtual patient from the virtual waiting line, event information will then be submitted to the DTM. At the time the real

[2]A sensor is a device that detects external information, replacing it with a signal that humans and machines can distinguish. In this work, the main purpose of the sensor is to detect patient arrival/departure events. Different types of sensors exist: USB sensors, wireless and Bluetooth sensors, embedded sensor modules, etc.

[3]DTM is the proposed online DT model responsible for monitoring patient pathways in real time.

#	Process flow element	Meta-model element	Description
1	In	Arrival Event	Illustrates the hospital entrance
2	Out	Departure Event	Illustrates the hospital exit
3	Delay with occupying resources	Activity	The activity may or may not necessitate the use of resources
4	Delay without occupying resources		
5	Medical operation		
6	Administrative operation		
7	Gateway	Gateway	Directs the flow of patients and determines the next activity
8	Move	Transition	Illustrates the walk path the patient may follow
9	Discrete event from the continuing connection		When the event handler receives an event from the real world, the event handler executes actions according to the event received
10	Continuous connection		
11	Instantaneous connection		
12	Connect elements		Connects the different elements of the process flow models together
13	Description		Describes the rules/ stochastic probabilities, in addition to some notes about the resources, among others
14	Group	Location	To describe the location of the activity (floor, room, etc.)

Table 3.4: Relation between the process flow model and the meta-model

patient arrives at the registration desk, the patient will be detected by the sensor attached to the registration desk, and the event information will be used to notify the DTM in order to move the virtual patient to the registration desk. Following this, when the patient finishes his/her registration, the sensor will detect the event of the patient leaving the registration desk and will inform the DTM to release the virtual patient from the virtual registration desk. After registration, the real patient will move to the exam room for treatment. At the time the patient arrives, the patient will be detected by the sensor attached to the door of the exam room, and the event information will be sent to the DTM, which will move the virtual patient to the virtual exam room. After the patient receives his/her treatment, the patient will leave the exam room. The sensor will detect the patient and inform the DTM that the real patient has left the exam room, so that the DTM will release the virtual patient from the virtual exam room. Finally, when the real patient arrives at the exit door, the sensor at the door will detect this patient and will inform the DTM that the real patient has left the hospital, and the DTM will remove the virtual patient from the model. To draw the process flow corresponding to this short use case, the following symbols will be used:

1. Symbol 1 to represent the entrance door of the hospital.
2. Symbol 8 to represent the patient's movement between the activities from the entrance to the exit.
3. Symbol 4 to represent a waiting line; the waiting line is considered a delay without occupying resources.
4. Symbol 5 to represent the medical activity, which in this example is the exam room.

5. Symbol 6 to represent the administrative activity (registration desk).
6. Symbol 2 to represent the exit door of the hospital.
7. Symbol 9 will be attached to the waiting line, the registration desk and the exam room activities; all of these are considered continuous activities, but we are interested in collecting the event at the beginning and at the end of the activity.
8. Symbol 9 will be attached to the entrance and the exit doors because the entrance and the exit also only represent one event.
9. Symbol 12 will be used to connect the different model elements.

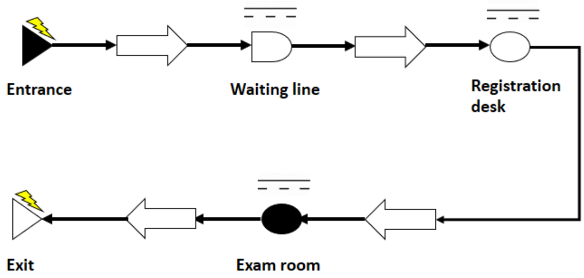


Figure 3.4: Process flow model

The process flow model in Figure 3.4 is an example of the enrichment model. For example, it is obvious that the waiting line does not consume resources. In addition, we can distinguish the different types of activities, such as the administrative activity from the medical activity. The way the model is connected to the real world is also clear, thanks to the different synchronization symbols that exist over the entrance, exit, waiting line, and exam room.

In short, this section shows the various modeling elements that will be used to draw an enrichment flow model; this is due to the need to draw up an enrichment model that includes as much knowledge as possible of the real patient pathways in our DT construction step in Section 3.4.1.4. The ultimate use of one of the commercial simulation tools is to transform this enhancement model into a DT model. FlexSim's discrete simulation of events was used for this work.

3.4.1.4 Construction

The first stage of the design phase is the construction step. The main objective of this step is to construct a process flow model, comprising all the information necessary for designing the offline and online DT models. The offline models in this phase will be used to validate the DT prior to connecting to the real world. These models are respectively called replay and prospective models.

The process flow model is the first model that shall be constructed to capitalize on, highlight, and formalize a set of knowledge about the patient pathways, such as the different pathways in the hospital, the different types of patient activities (medical, administrative, waiting, and moving), the distances the patients may travel from one activity to another, the theoretical speed of patients, and the duration of each activity (as a static distribution). As shown in Figure 3.5, in order to design the process flow model, the historical data from the Hospital

Information System (HIS), the statistical analysis, the theoretical measurements, and the developed designer elements that have been discussed in Section 3.4.1.3 will be used.

After developing the process flow model, the offline simulation models will be developed. As shown in Figure 3.5 for Step 3, the output knowledge from the process flow model will be used as input for the creation of the offline simulation models. These models are called **Replay model** and **Prospective model**.

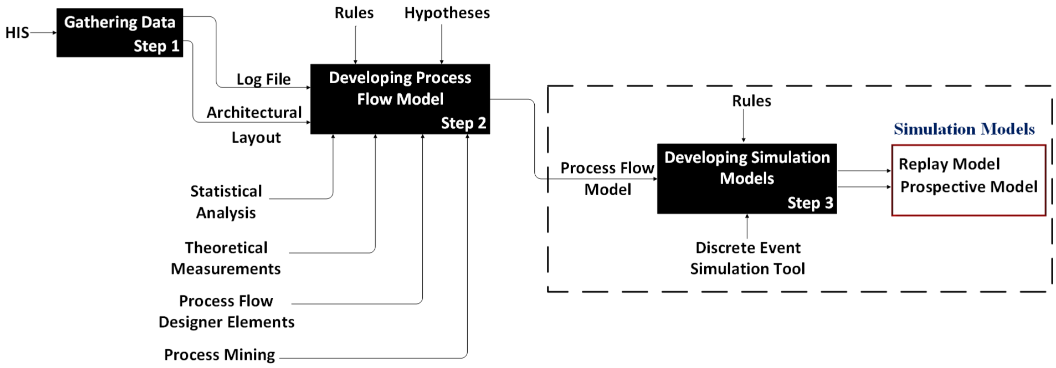


Figure 3.5: From Hospital Information System to simulation models

In Figure 3.5, in addition to the log file events, the hospital's architectural layout might be used as inputs to the Step 2, which is developing a process flow model (an enriched model) to convert the aforementioned inputs into a process flow model. Various mechanisms can be applied, such as using statistical analysis on the log file, for example, to answer a variety of queries, including:

- What are the different activities visited by the patient in the hospital?
- How many patients visit the activity called X?
- How many patients enter/leave the hospital during a specific period of time?

Based on the theoretical knowledge regarding patient speeds and the various measures that are illustrated in the architectural layout, the average distance that the patient travels from entrance to exit can be determined.

From the architectural layout we have knowledge of the structure of the hospital building (rooms, floors, etc.) as well as the location of each activity, the fixed resources, the distance the patient may need to travel between one activity to another (walk path). Moreover, using a process mining tool can help to find the different patient pathways, the average number of patients following each pathway, the behavior of the hospital, etc.

To construct a rich process flow model, the information obtained from statistical analysis, a process mining tool and the architectural layout will be used to select a suitable modeling element (process flow elements in Section 3.4.1.3). To do this, several hypotheses and rules could be utilized. The following hypothesis, for example, could be used:

1. Any discovered patient activity shall be labelled as (waiting activity, medical activity, administrative activity); the designer elements in Section 3.4.1.3 will be used to depict these activities:
    - Any waiting line activity can be considered a type of delay without occupying resources. Symbol 4 will be used to illustrate this type of activity.

- Any waiting room activity can be considered a type of delay with occupying resources. Symbol 3 will be used to illustrate this activity.
- Any medical operation can be considered a medical activity. Symbol 5 will be used to illustrate this.
- Symbol 6 will be used to illustrate any discovered administrative activity.

2. Moving elements in the process flow will be used to define the different walk paths the patient may follow from one activity to another.

In addition to the above hypothesis, different rules could be used such as:

- Each activity must have a duration time; the simulation model needs the duration for the activity to run, except in the case of events. For example, the activity will be started and finished based on two events (start and end event).
- The walk path that connects two activities must be clear: the simulation model needs to know the different pathway the patient may follow.
- The speed of the patient could be selected approximately. For example, 1 m/s; the patient in the simulation moves at a deterministic as well as a stochastic speed based on need.

The replay model is an offline simulation model that is used to replay the constructed log file of events. In this work, the FlexSim DES tool has been used to build this simulation model. This model must run using the log file as input data and thereby will behave nearly the same as the real world did in the past. Different indicators can be used to achieve this goal. For instance, a **virtual** patient must be created at the entrance of the simulated hospital to correspond to the same event recorded in the log file for the arrival of the **real** patient in the hospital. The duration time for each activity for each patient must be equivalent to the same duration time that was calculated from the log file. In addition, the virtual patient must follow the same pathways as those discovered in the log file, etc.

The main reasons for the construction of the replay model are: (1) in order to be able to replay the various paths that have been taken in the past, the same activities as in the real world must be implemented in the simulation model. This will help to understand hospital behavior based on past events; (2) checking the conformance of the model with the log file. For example, checking whether or not the simulation model differs/diverges from past events; (3) checking the structure of the model; and (4) discovering process weaknesses (bottleneck activities, resource utilization, etc.) before improving the model.

The prospective model is an offline virtual simulation model that simulates the process by using distributions of random variables (patient arrivals, duration of each activity, walking speed, decision rules, etc.) rather than the replay of a log file. This model used patient profiles to simulate the future. In other words, this model uses **static** random distributions that are calculated from the log file to simulate the future. One of the major differences between the replay model and the prospective model is that the execution of the replay model should follow the events recorded in the log file, while the execution of the prospective model is based on random distributions calculated from the log file. In other words, the replay model can only replay past events, while the prospective model can simulate past and future events, as shown in Figure 3.6.

3.4.1.5 Validation

The second step of the design phase is to validate the developed replay and prospective models before transforming them into a DTM and a DTP. In fact, these two models are

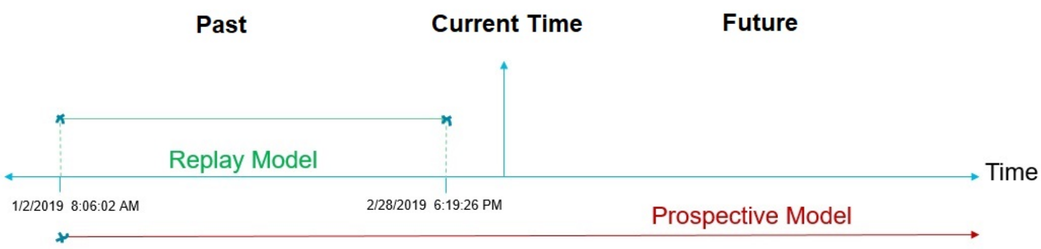


Figure 3.6: Time line: the difference between the replay model and the prospective model (Karakra et al., 2020)

considered the core of the DT. The replay model will be transformed into a DTM to monitor patient pathways in near real time, and the prospective model will be transformed into a DTP to predict the near future of these pathways. Because the replay model is built using the events extracted from the hospital information in a log file, the replay model will be validated using the same log file. Moreover, the prospective model can be validated using the valid replay model.

To perform the validation process, a set of common indicators, e.g., Key Performance Indicators (KPI) and a threshold value are selected. These indicators will be used for the comparison between (1) the events generated from the replay model with the events stored in the log file to validate the replay model, and (2) between the events generated from the replay model with the events generated from the prospective model to validate the prospective model. The threshold value will be used to verify whether the model is valid or not. However, the comparison indicators and the threshold value could be selected by the hospital manager and/or experts in the hospital. Algorithm 1 depicts a suggested algorithm that could be used in the validation process.

Algorithm 1 Validation Process (Karakra et al., 2020)

Require: $maxRange$, $minRange$, $passTestCounter$, $indicatorsLength$
function IsValid($IndicatorsList$, $ReferenceModel$, $threshold$, $Model$)
  $passTestCounter \leftarrow 0$ ▷ Number of indicators pass the validation test
  $indicatorsLength \leftarrow IndicatorsList.length$ ▷ Number of indicators to validate
  for each $indicator \in IndicatorsList$ **do** ▷ To cover all of the indicators in the list
    $maxRange \leftarrow ReferenceModel.indicator + (threshold * ReferenceModel.indicator)$
    $minRange \leftarrow ReferenceModel.indicator - (threshold * ReferenceModel.indicator)$
    `if Model.indicator< maxRang and Model.indicator > minRange then`
      $passTestCounter \leftarrow (passTestCounter + 1)$ ▷ All indicators must be confirmed
    `end if`
  end for
  `if passTestCounter = indicatorsLength then`
    return 1 ▷ Valid model
  `else`
    return 0 ▷ Not valid model
  `end if`
end function

In Algorithm 1, the reference model is the model/file that will be used as the base to compare with. For example, the log file will be considered the reference in the comparison between the replay model and the log file, and the replay model will be considered the reference model when comparing the replay model with the prospective model.

Each indicator must pass the test in Algorithm 1 to indicate whether the model is valid. This means the percentage of validity is 100%. If one or more indicators do not pass the test, this means the model is not valid unless the committee decides that the model is valid for any reason. For example, they can decide the model is valid if the percentage of the validity is 80%, which can be calculated by dividing the number of valid indicators by the total number of indicators, multiplied by 100% .

3.4.2 Ramp-up phase

This phase is considered an online phase in which the offline simulation models (replay and prospective) created during the design phase will be transformed into a DTM and a DTP, where real-time events will be used as input for these models.

Initially, the replay simulation model will be converted into a DTM in this phase. This means that instead of running the simulation model on the basis of log file events, the simulation model will run on the basis of the events collected in real time by sensors, as illustrated in Figure 3.7. In the same way, the prospective model will be transformed into a DTP. Instead of running the DTP on the basis of **static** random distributions, this model will be run on the basis of empirical **dynamic** distributions, as shown in Figure 3.8. Dynamic distributions mean that the shape of the distribution will be updated automatically according to the continuous update of the collection of events, which means the shape of the distribution may change. Thus, the shapes of the dynamic distributions can be different than the shapes of the static distributions for the past period. Figure 3.9 explains the distinction between static and dynamic distributions in greater detail.

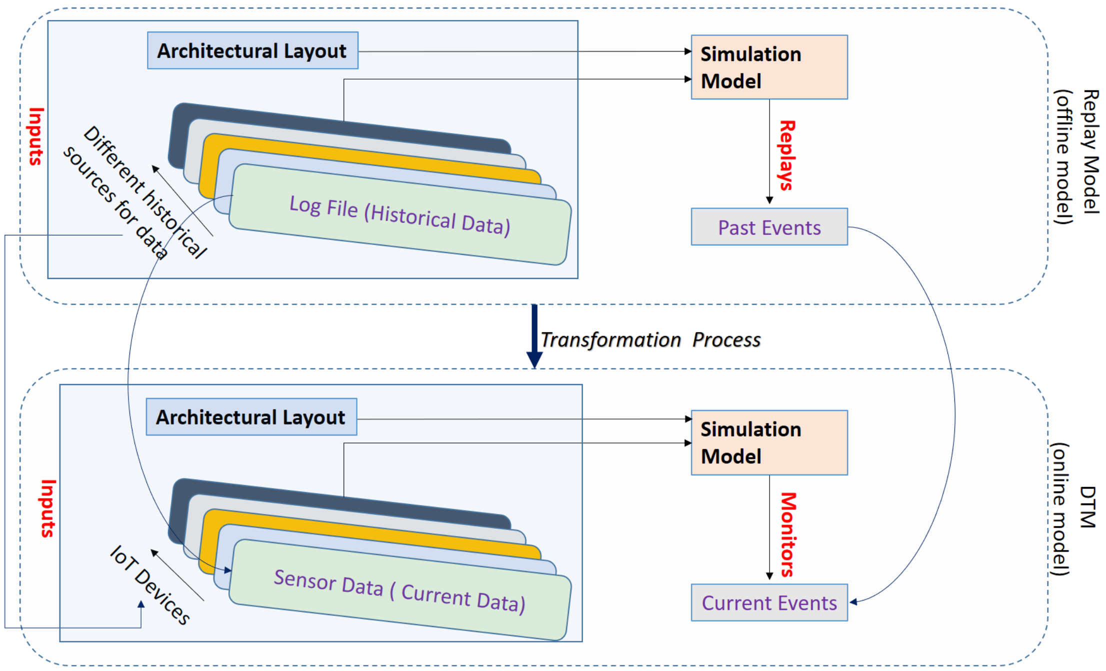


Figure 3.7: The transformation process: from the replay model to a DT for monitoring

Figure 3.7 depicts the process of transforming the replay model into a DTM. The transformation process is summarized by the following steps:

- Historical data in the replay model will be transformed into sensory events in the DTM. To accomplish this goal, various IoT devices (e.g., sensors) can be installed along the real patient pathway to capture real-time data to feed the DTM, and a copy of this data will be kept in the storage area (e.g., database) to be used for other purposes afterwards.

- Instead of running the model in an empty state, the model will be initialized with the current state of the hospital in the DTM. To achieve this objective, the sensory data stored in the repository area can be used.

- Instead of using the simulation clock (a variable giving the current value of simulated time) in the replay model, the real world clock, or what is known as the **wall clock**, will be used in the DTM. To manage this need, the simulation model will execute at the same rate as the real world wall clock time.

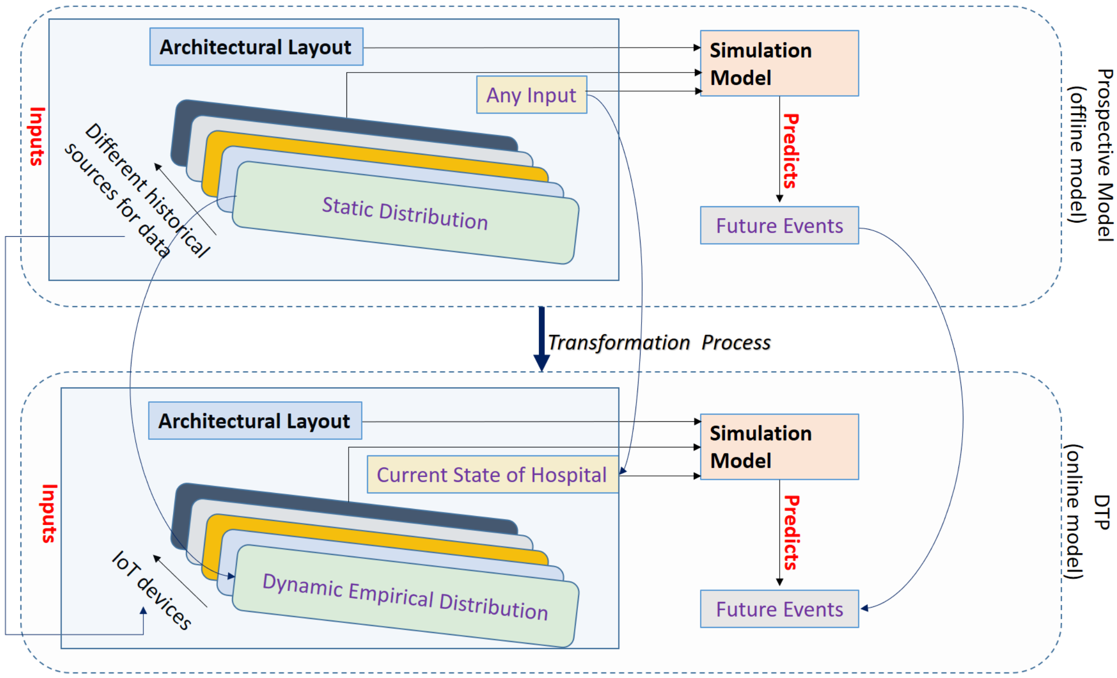


Figure 3.8: Transformation process: from the prospective model to a DT for predicting

Figure 3.8 summarizes the process of transforming the prospective model into a DTP; the prospective model starts with inputs given from the decision-maker. Based on the static distributions (the mathematical functions discovered from the log file); this model will be started in order to anticipate the future. On the other hand, the input of the DTP is the current state of the hospital, and the model will be started using the updated dynamic empirical distribution by the sensors. The transformation process from the prospective model into the DTP is summarized by the following steps:

- Historical data in a prospective model will be transformed into the sensory events in the DTP. This means that the data captured by the sensors will be used to update the historical data. This need can be accomplished with the help of the installed IoT devices and the storage area. In other words, instead of initializing the model from the user, the model will be initialized by the current state of the hospital in the DTP. Two approaches can be considered in achieving this goal: (1) Using the DTM to clone the current state of the patient pathways and using that clone as an input for the DTP to initialize it immediately before beginning to anticipate the future. This approach is dependent on the software's capability to replicate the state from one model and load it into another model. (2) The data in the storage area can be utilized to initialize the DTP immediately before starting to predict the future. There are some advantages and limitations for each of these approaches that will be discussed in Chapter 4.

- The static distribution in the prospective model will be updated to the dynamic empirical distribution in the DTP. This means that the shape of the distribution could change from time to time based on the nature of the data. This objective might be achieved behind the scenes by updating the distributions each time a real world event is detected, and these distributions would then be kept in the storage area. The most recent empirical distribution will load from the storage area into the model immediately before starting to anticipate the future when the DTP needs to be started.

> Updating the empirical distribution consumes resources such as time and space. This is determined by the volume of data, the processing algorithms and the speed of the processing machine, in addition to the mathematical technique used to calculate these types of distributions. These distributions must be updated behind the scenes on a different server and they must be updated every time an event is detected. The most recent updated distributions must be kept in the storage area, and they will be loaded into the model immediately before the time the DTP is required to be run.

To summarize, there are some differences between traditional offline simulation and the DT models, (DTM and DTP), in terms of business (usage) and technical aspects (construction) as follows:

- Business aspects:
  - The DTM has a continuous connection with the real world, but the model is updated at each detected event. In fact, the objective of this model is to show the current state of the real world in near real time. Moreover, the DTP has a semi-connection with the real world, which means that the model is connected with the real world to get the current state of the real world, as well as the most recent updated distributions. After that, the model will continue running in an offline mode to predict the near future. In contrast, the offline simulation is not linked/connected to the real world.
  - The offline simulation starts with an empty state, whereas the DT models start with a non-empty state (the model starts in a state similar to the real-world state).
  - Running the DTP is based on current data and updated dynamic empirical distributions, whereas running the offline simulation for predicting is dependent on user input and the model is run based on static distributions.
  - The DT can control the real world by using the different actuators and decision-makers in the real world, and it can provide feedback (e.g., alarm messages) to the real world, whereas the offline simulation is used to assess different scenarios, such as what-if scenarios. In this work, the DTM is used to provide alarm messages, and the decision-maker can make decisions according to the messages received. However, the different usage for the DTM and DTP will be discussed in Section 3.4.3.
- Technical aspects
  - The DT models include components that allow them to receive real-world events.
  - The DT models include components that enable them to send feedback to the real world.
  - The DT models include components that enable them to perform actions in response to events received.

– The DT models contain components that allow them to control and/or manage the real world via actuators and decision-makers. More details about the technical aspects will be illustrated in Section 3.4.3.

Traditional simulation tools (for running offline prospective simulations) lack all of the aforementioned components for building an online model.

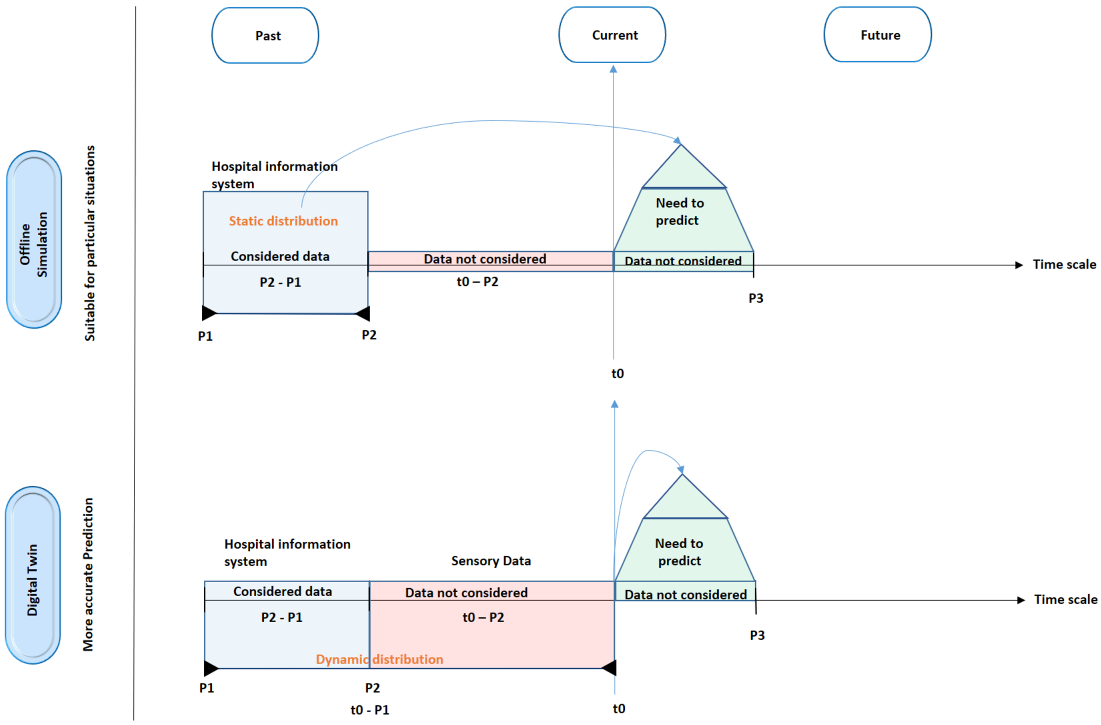


Figure 3.9: Static distribution versus dynamic distribution

Figure 3.9 points out the differences of the offline simulation and the DTP in terms of probability distributions (static and dynamic distributions). In the offline simulation part, the figure is divided into three parts according to the time scale (past, current, and future), and the past is divided into two parts according to the duration of the collection of the data (duration of the interval [P1,P2] and duration of the interval [P2,t0]). Normally, in the static distribution, the data that will be used to simulate and to predict the behavior of the system is the data that is collected only in the interval [P1,P2], whereas the data that is collected in the duration of [P2,t0] is ignored. Sometimes, if the interval [P2,t0] is a long period and there are many significant actions happening in this period while at the same time, all of these actions are ignored, inaccurate predictions for the future could be made. In this case, the prediction model will not be valid for predicting the future, or it may be suitable for particular scenarios that happen only in the past, specifically in this interval of time [P1,P2]. In the DTP, the data used for predicting the future is normally the data that exists in the interval [P1,t0], which includes a rich level of information compared with the previous interval [P1,P2]. According to (Grolinger et al., 2016), a massive amount of data leads to more accurate prediction and better decisions. The idea behind collecting real-time data in our case is related to: (1) enriching our past data, and (2) adding the current data to the past data leads to including different scenarios happening currently for which there is no information about them in the past. Section 3.4.3 details the procedure of deploying the DT models, (DTM and DTP), in the real world. The different scenarios for switching from the DTM to the DTP have been illustrated in the same section.

3.4.3 Running phase

The running phase is the final phase of the DT development life cycle, where the DT models will be enacted in a real hospital. At this phase, the DT models will be set up, configured and customized to be used in the real world hospital. At the beginning, the DT models, (DTM and DTP), will be set up in the hospital servers, then must be configured according to the hospital's states and behaviors. For example, the DTM must be initialized with the current state of the real hospital. For instance, if the real hospital had one patient at the RD, three patients in the WR, etc., the DTM needs to be in exactly the same state. To reflect reality, the DTM clock will be synchronized with the real world wall clock. Afterwards, the DTM must be run in parallel with the real world at near real time. For example, when an event occurs in the real world, the same event must occur in the DTM at near real time. If the DTM detects an unexpected event, such as a delay, a missed appointment, crowding, etc., the DTP will run to predict how the event will affect the real hospital. After that, a proactive decision can be made to reduce the impact of this event.

Figure 3.10 represents a kind of a proposal for connecting the virtual world with the real world. In the real world, various elements can exist, such as tags (e.g., RFID tags, virtual tags, etc.), sensors (RFID readers, etc.), and decision-makers. In the virtual world, a DTM and a dashboard that represents the DTM statistics can exist. At the middleware level (which is a kind of communication environment for bridging the gap between the real world and the virtual world), different types of elements may exist to facilitate communication between the DTM and the real world, such as processing tools, brokers, and data storage. A description for the different elements that can be used to connect the DTM with the real world is illustrated in Table 3.5.

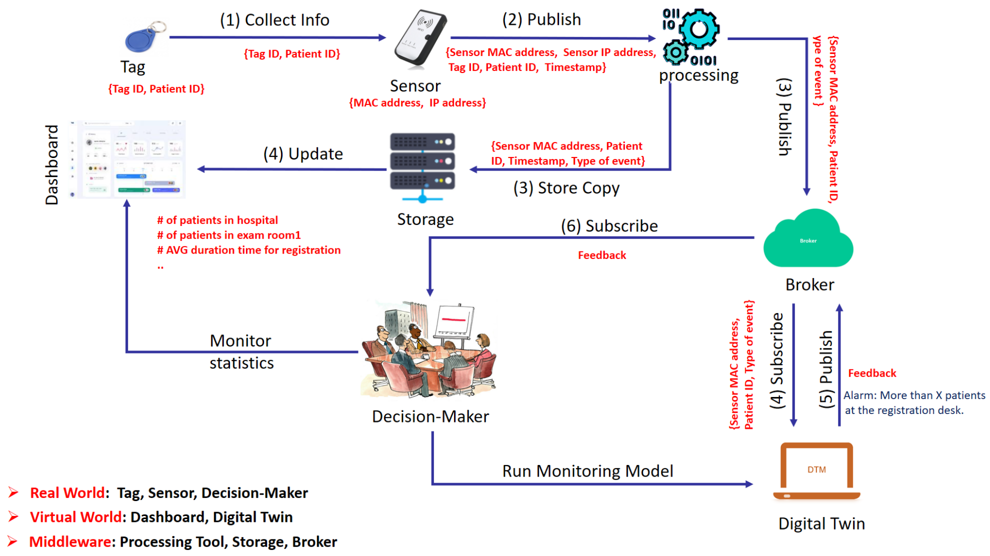


Figure 3.10: Digital twin for monitoring: the monitoring process

Element	Description
Tag	Stores different types of data such as tag IDs and patient IDs. Used as a reference for the patient.
Sensor	Can be located by using MAC (Media Access Control) addresses and IP (Internet Protocol) addresses. Used as a reference for the activity.
Processing tool	Used to transform the data into the required structure.
Storage	Used for data storage (past, current, and predictive outcomes).
Dashboard	Used to show different statistics regarding the past/current and future state of the patient pathways of the hospital.
Decision-maker	The one who runs the DT and makes the decisions according to the feedback from the twins.
Broker	A kind of centralized server that may use the Publish/Subscribe pattern that provides a framework for exchanging messages between publishers and subscribers. The publisher is the sender of messages and the subscriber is the receiver of the topics to which they have subscribed.

Table 3.5: Proposed elements for connecting the DTM with the real world

As shown in Figure 3.10, the steps listed below demonstrate the connection between the DTM and the patient pathways in a real world hospital.

1. At the time the patient arrives at an activity, a sensor attached to this activity detects the tag associated with the patient.

2. Different information can be collected from the detected tag thanks to the detection sensor, such as the {Tag ID, Patient ID}.

3. The sensor publishes a set of data for the local processing tool. This data includes data from the detected tag as well as data from the detection sensor itself, such as {Sensor Mac Address, Sensor IP Address, Tag ID, Patient ID, and Timestamp}.

4. The local processing tool itself formats the received data into the different required structures. For example, the data that follow this structure {Sensor MAC address, Patient ID, Timestamp, Type of Event} will be stored in the data storage, and the data that follow this structure {Sensor MAC Address, Patient ID, Type of Event } will be published in the broker. The second structure is the one that is proposed for feeding the DTM with data.

5. Because the DTM is one of the broker's client subscribers, the DTM receives the published data from the broker, and the format of this data is as follows: {Sensor MAC Address, Patient ID, Type of Event}. The sensor **MAC Address** will be used by the DTM for locating the activity and the place where the sensor is attached (e.g., entrance door), whereas the **Patient ID** for locating the real patient and **Type of Event** will be used by the DTM to know which action must be applied to the virtual patient inside the model (e.g., create patient, block patient, etc).

6. In case an unexpected event is detected by the DTM, there is a possibility of the DTM publishing feedback (e.g., an alarm message) that explains the type of detected event. In this case, the DTM will be considered a publisher.

7. Because the decision-makers subscribe to some topics on the broker, they will receive the published alarm messages. Based on the received message, the decision-makers can make a decision. Indeed, they can monitor the dashboard to see the current statistics in the hospital (e.g., number of patients in hospital, number of patients in exam room 1, the average duration time for registration, etc.) and they can run the DTP, as will be discussed below, to anticipate the future and see the impact of this event on the real hospital and to see what the possible solutions are for minimizing the impact of this event.

8. Finally, the decision-maker will make a decision corresponding to the feedback received from the DTP.

Three different approaches are proposed in this research work to run the DTP as follows:

1. Reactive approach: in case of unexpected events happening in the real world, the decision-maker will run the DTP to anticipate the near future and see the impact of this event on the real hospital, and to see what the possible solutions are for reducing the impact of this event. Based on this, the decision-maker will make a decision.

2. Proactive approach: this approach is considered a periodic approach because in this approach, the decision-maker runs the DTP periodically, for example, every t time unit, to anticipate the near future. If the DTP detects a deviation in the near future, the decision-maker will try to make a proactive decision to reduce or mitigate the impact of this deviation before it happens.

3. On-demand approach: in this approach, the decision-maker can choose the DTP for whatever reason; for example, the decision-maker wants to have a proactive overview of a certain activity, or a proactive overview of the state of the hospital after three hours.

Figures 3.11, 3.12, and 3.13 illustrate the DTP approaches. Figure 3.11 illustrates the steps of running and utilizing the DTP to predict the future based on the reactive approach. These steps are summarized as follows:

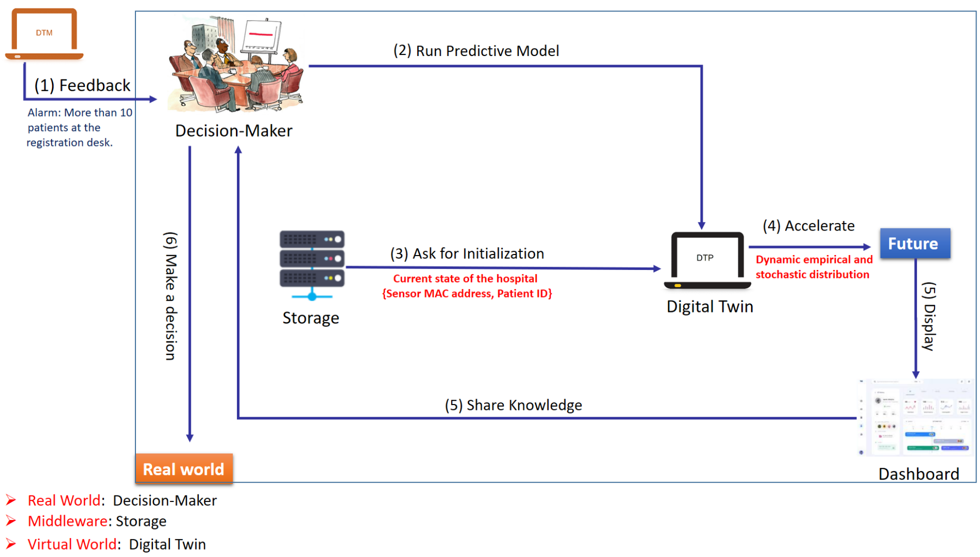


Figure 3.11: Digital twin for predicting: the reactive approach

1. The decision-maker will receive feedback, such as an alarm message from the DTM that mentions that there are unexpected events; for example, more than 10 patients at the registration desk.

2. The decision-maker will run the predictive model to anticipate the near future to see the impact of this event on the real hospital.

3. The DTP will be initialized with the hospital's current state using the existing data storage.

4. After the initialization process, the DTP will run at a faster speed than the real world speed to predict the future.

5. Feedback from the DTP on the future will be displayed on the dashboard. For example, the impact of the event, and what the possible solutions are for minimizing the impact of this event. This knowledge will be shared with the decision-maker.

6. Finally, the decision-maker can make a decision according to what has been received from the DTP.

Figure 3.12 shows how to run and use the DTP to predict the future based on a proactive approach. These steps are summarized as follows:

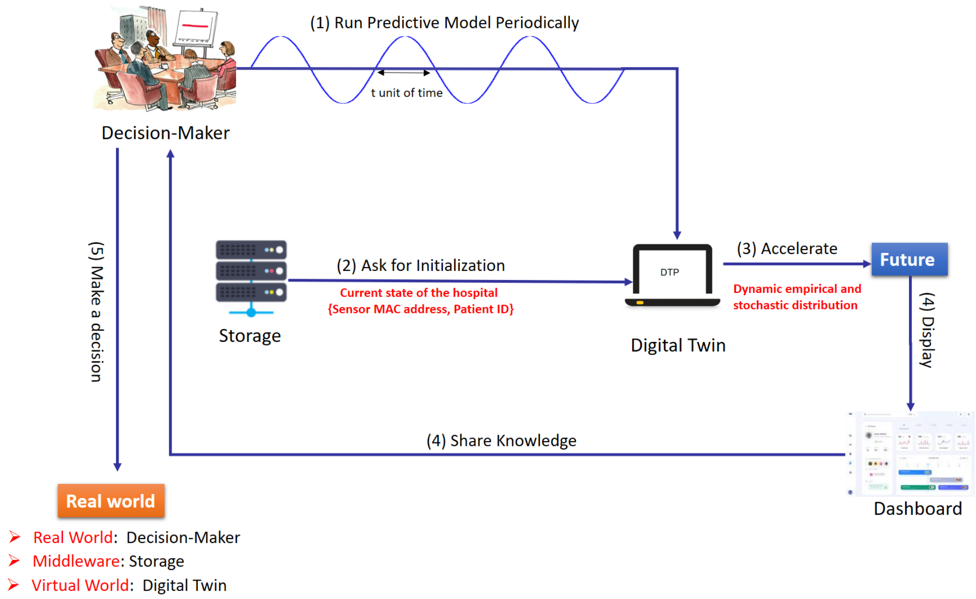


Figure 3.12: Digital twin for predicting: the proactive approach

1. Similar to steps 2–4 in the reactive approach, except that the decision-maker will run the DTP periodically at each t unit of time; for example, every two hours (this does not follow an event-driven approach).

2. Proactive feedback from the DTP on the future will be displayed on the dashboard. This feedback will be shared with the decision-maker.

3. If there is an unexpected event that will happen in the future, the decision-maker can make a proactive decision to minimize or mitigate this event before it happens.

The main difference between the reactive approach and the proactive approach is that in the reactive approach, the event occurs in the real world and the decision-maker will try to reduce its impact on the real hospital in the near future. In a proactive approach, a deviation will happen in the future (in the virtual world, not real world), and the decision-maker will try: (1) to prevent it before it happens in the real world if possible, or (2) to reduce or mitigate its impact in case it does happen.

Figure 3.13 demonstrates how to run and use a DTP to predict the future on the basis of the on-demand approach.

1. Similar to steps 2–4 in the reactive approach, except that the decision-maker runs the DTP based on his/her demand. For example, he/she runs the predictive model to see the state of an activity called X after t unit of time.

2. Overview feedback from the DTP about the future of activity X will be displayed on the dashboard, and this feedback will be shared with the decision-maker.

3. Based on the feedback received, the decision-maker can make a decision to improve the selected activity.

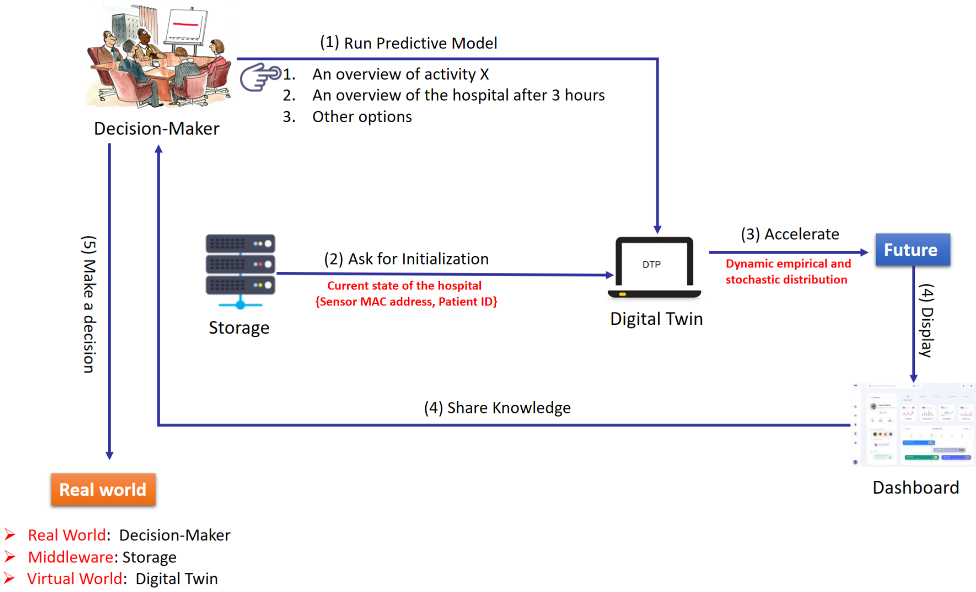


Figure 3.13: Digital twin for predicting: the on-demand approach

It should be noted that in addition to the DTP, the decision-maker can use another type of DT model that we can call the Digital Twin for Optimization (DTO). This DT can search through thousands of different ways to allocate resources and, for each one, thousands of different possible future outcomes to find the best set of options (Carson et al., 1997). These options will provide the decision-maker with a much better picture of the choices that can be used to optimize the patient pathways. For example, suppose the decision-maker decides to predict the state of the hospital after 2 hours, and the results provided by the DTP do not match the decision-maker's objective. In this case, the decision-maker can run the (DTO), whose role is to find the best input variable values from among all possibilities to achieve the decision-maker's goal/target, thanks to a connection to an optimization algorithm or something similar. However, this type of DT is not included in the scope of this work.

This section illustrates the different components that can be used to deploy the DT models, (DTM and DTP), in the real hospital. Different components can be utilized depending on the hospital's organizational structure, the hardware or software it uses, and so on. The various uses for the DTP have been demonstrated, and the various types of events that this twin can detect have been explored. The key aspects of this chapter will be summarized in Section 3.5.

3.5 Conclusion

To sum up, this chapter helps to lay the groundwork for a new framework called HospiT'Win. It defines the procedure for designing and developing a DT for real-time monitoring and the prediction of the patient pathways.

This chapter proposed two meta-models and process flow modeling elements to help the designer and the developer of the DT. One meta-model corresponds to the real patient pathways in the hospital. It highlights the various real-world patient pathway elements and the relations between them. This meta-model will help us in understanding how the patient pathways work in the real world as well as their main components. In fact, understanding the real patient pathways makes it easier to develop a DT for these pathways; the DT is considered a dynamic virtual representation for these pathways.

The second meta-model corresponds to the DT of patient pathways in the hospital. This meta-model provides knowledge about the different requirements that are needed to design and develop the DT models. The relation between the two meta-models is detailed in this chapter. Furthermore, the process flow modeling elements that have been developed will help to capture different knowledge from the real world patient pathways into one model called the "enriched process flow model". This model will reduce the time needed by the DT designer to design the proposed DT models. In fact, the proposed modeling elements can be considered rich elements that capture different knowledge according to what the DT designer needs. For example, the different pathways the patient may follow, the duration for each activity, the different types of activity (with/without resources) and the way of connecting the activities with the real world, in addition to different information.

A global development methodology for building the proposed DT models, from the designing to the running phases, has been provided. This methodology is divided into three phases and four steps. The phases of this methodology are the design phase, the ramp-up phase, and the running phase. The steps of this methodology are construction, validation, transformation, and deployment.

The construction and validation steps are responsible for developing offline simulation models and validating them before transforming them into DT models. The transformation step is responsible for transforming the offline simulation models into a DTM and a DTP. Finally, the deployment step is responsible for connecting and running the DTM and DTP models in a real hospital.

The different requirements that are needed to design and develop the proposed DT models, starting from collecting the data from the real world and running them, have been described in detail. Also, the main components for the DTM and DTP models have been illustrated. The difference between the developed models and the way of usage is explained in detail.

This chapter has illustrated three approaches for switching from DTM to DTP: a reactive approach, a proactive approach, and an on-demand approach. The difference between these approaches has been described. In fact, there are several important concepts that must be considered in addition to what has been described in this chapter about the DTM and the DTP: As examples, the initialization of the DTM or the DTP with the current state of the real world, or furthermore, synchronizing the DTM with the real world and accelerating the DTP to anticipate the future. All of these concepts will be clarified in detail in Chapter 4.

4

Initialization and Synchronization of the Digital Twin with the Real World

4.1 Introduction

Chapter 3 discussed a proposed methodology for designing the proposed Digital Twin (DT) models. Certain phases and their steps have been explained. These steps begin with the collection of historical data and culminate with the execution of the DT models in the real world. Two main concepts that are mentioned in that chapter are initialization and synchronization. Initialization means starting the DT models with the current state of the hospital. In other words, instead of running the model in an empty state, the model must reflect the current behavior of the real world when it starts running. Synchronization means that the DT must be connected and running in parallel with the real world. Each time an event is detected in the real world, this event must be reflected in the DT in near real time. In fact, these two concepts were not covered in depth in that chapter, as Chapter 3 did not go into detail on the algorithms that are used to initialize and to synchronize the DTs with the real world.

According to the literature reviewed in Chapter 2, initialization and synchronization of the DT models with the real world are among the top issues that must be resolved. As a result, the aims of this chapter are twofold: first, initialization, and second, synchronization. In the initialization segment, the chapter details how to initialize the proposed DT models (Digital Twin for Monitoring (DTM) and Digital Twin for Predicting (DTP)) with the current state of the real patient pathways. To explain this part, proposed algorithms are used. Other algorithms in addition to a developed modeling graph (GRAFNet) for describing a discrete event system are used in the synchronization segment to demonstrate the main concepts behind synchronizing the DTM of patient pathways with the real patient pathways at each event detected in the real pathways.

4.2 Definitions and Assumptions

In a real hospital, there are many situations where a patient arrives at an activity (e.g., medical examinations and operations, diagnoses, etc.) and the healthcare practitioner is unavailable because he or she is on a call, or waiting for resources which aren't yet available to start the activity, or has something urgent to finish before seeing the patient. However, the activity is not considered to have begun until the practitioner initiates it. These examples demonstrate different scenarios of the presence of a patient within an activity that has not been started for whatever reason. Similarly, technical issues may arise within the hospital, such as when a sensor fails to detect the patient as he or she enters or exits the activity or hospital. This issue has no bearing on how the activity is started, but it may have an impact on how it is monitored and controlled. According to all of this, a set of definitions and assumptions has been developed to illustrate how to initialize and synchronize the proposed DTs with the real-world patient pathways in the hospital.

Three terms will be used throughout this chapter: **event**, **state**, and **activity**. For this reason, a clear and precise definition of these terms is needed, as illustrated below:

> An event is a specific instant in time that marks the start or the end of an activity, and this event consumes neither time nor resources, whereas an activity is the actual execution of the task, which takes time and resources to complete. The duration time of the activity can be calculated from subtracting the timestamp of the start event from the timestamp of the end event.
>
> The state is a set of variables or parameters that describes the status of an activity at any given time. For example, an activity is busy, idle, etc. The state is changed upon the occurrence of an event. For example, when the patient leaves the activity, the state of the activity will change from busy to idle, and so on.
>
> The state of the patient pathways is a set of variables/parameters that describe the situation of all patient pathways at any given time (e.g., three patients in the waiting room, one patient at registration, all of the exam rooms are busy, etc.).

Because we have not yet connected our DT to the real world hospital, and to serve the purpose of this work, some assumptions have been developed as a first step:

A1 *We consider that the medical and administrative activities start at the time the patient arrives at the activity, and the activity finishes when the patient leaves.*

Example: At the time the real patient arrives at the activity (e.g., medical activity), the twin's activity in the DT is considered to be started. In the real world, sometimes the patient arrives at the activity (e.g., medical examination room) while the doctor is still outside. In this research, we assume that the DT's activity begins at the same time the real patient arrives, and that all of the resources needed to begin the activity are available even if the real doctor has not arrived. The same assumption has been applied to the patient's departure. For example, the activity will finish at the time the patient leaves.

A2 *Sensors that are used to perform the synchronization between the virtual and the real worlds are considered to be operationally sound, with no technical issues.*

Example: At the time when the patient arrives at/leaves any real-world activity, we consider that there is a sensor attached to this activity, and this sensor has to detect the event of this patient arriving at/leaving the activity, and once this sensor has captured the event information about the detected patient, this information must be published in (sent to) the DT. Thus, the DT can synchronize the virtual twin with the real patient.

A3: There are two kinds of events in this work:

1. Arriving at the activity or hospital, which represents when the patient arrives at the activity or the hospital entrance,
2. Leaving the activity or hospital, which represents when the patient leaves the activity or the hospital exit.

4.3 Initializing the Digital Twin with the Current State of the Real World

This section aims to address the question of how to initialize the DT with current knowledge about the state of the real patient pathways in the case of the monitoring model (DTM) and the predicting model (DTP). This question raises two critical issues for consideration:

- What data should be gathered from the real world in order to perform the initialization?
- Due to using Discrete Event Simulation (DES), is the data gathered from the real world going to be helpful in performing the initialization?

In this work, initialization refers to the process of starting the model in a "non-empty or idle state", but in the same state as the hospital's current patient pathways (Bergmann et al., 2011). For example, if there are three patients in the waiting room, one in the exam room 1, and none in the waiting line, the DT must begin with three patients in the virtual waiting room, one in the virtual exam room 1, and none in the virtual waiting line. In reality, this differs from conventional or traditional simulation, in which the simulation model begins in an empty or idle state (Hanisch et al., 2005). To clarify the initialization phase, we have identified the following four related sets of information: R, IDsR, V, and IDsV:

1. R represents a set of activities in the real hospital.
2. IDsR represents a globally unique set that is used to keep track of patient IDs in the real hospital.

   Based on (1) and (2), the following basic notions are used:

   - r describes a single real world activity, where $r \in R$,
     and r={activity name, current number of patients in the activity, IDsR }.
   - $\mid R \mid$ returns the number of activities in the set R.
     - Example: the set R below represents different activities in the real-world hospital as follows:
       * R= {(WL,3,{123, 234, 345}), (RD, 1, {678}), (WR, 2,{490, 530}), (ER1,1,{773 } }), (ER2, 0,{}) }
       * r1 = { WL, 3, {123, 234, 345} }

        * r2 = { RD, 1, {678} }
        * r3 = { WR, 2, {490,530} }
        * r4 = { ER1, 1, {773 } }
        * r5 = { ER2, 0, {} }
        * $\mid R \mid$ = 5 activities

3. V represents a set of activities in the virtual hospital model (DTM or DTP).

4. IDsV represents a globally unique set that is used to keep track of patient IDs in the virtual model.

   Based on (3) and (4), the following notions are used:

   - v describes a single virtual-world activity, where $v \in V$, and v={(activity name, current number of patients in the activity, IDsV)}.
   - $\mid V \mid$ returns the number of activities in the set V. Note: $IDsR \equiv IDsV$ (should be the same immediately after the initialization process), and $\mid V \mid \equiv \mid R \mid$; the number of activities in V must be equivalent to the number of activities in R before initializing the DT even if the number of activities in the real world is sometimes greater than the number of activities in the DT. In other words, the DT has to include all of the activities that need to be tracked/monitored in the real world. Assume there are ten activities in the real world, but we just need to track five of them. In this case, $\mid R \mid$ =5 not 10.
     - Example: the set V below represents the virtual-world activities before initializing the DT:
        * V= {(WL, 0,{}), (RD, 0,{}), (WR, 0,{}), (ER1, 0,{}), (ER2,0,{})}.
        * v1 = {WL, 0,{} }
        * v2 = {RD, 0,{} }
        * v3 = {WR, 0,{} }
        * v4 = {ER1, 0,{} }
        * v5 = {ER2, 0,{}}
        * $\mid V \mid$ = 5 activities
     - Following the DT initialization, R and V must be as follows:
        * R= {(WL,3,{123, 234, 345}), (RD, 1, {678}), (WR, 2,{490, 530}) ,(ER1,1,{773 } }), (ER2, 0,{}) }
        * V= {(WL,3,{123, 234, 345}), (RD, 1, {678}), (WR, 2,{490, 530}) ,(ER1,1,{773 } }), (ER2, 0,{}) }

According to these developed sets, Algorithm 1 (Karakra et al., 2020) depicts the process of initializing the DTM with the same state as the real world (real patient pathways).

Technically, a database table can be used to implement the two sets R and IDsR. As such, the header of this table could be: {TableID: int, PatientIDs : text, PatientLocations: text, ...}. PatientIDs are patient IDs, whereas PatientLocations are the IDs/MAC addresses of the sensors that detected the patient when he or she arrived at/left the activity/hospital.

The first step of Algorithm 1 is to call a **reset procedure** in the simulation tool. This procedure is responsible for pulling the patient information from the database and creating virtual patients in the DTM model corresponding to their information pulled from the database, such as their IDs and in which activity they are. The location of the activity can be determined with the Media Access Control (MAC) address assigned to each sensor attached at each activity. The loop in Algorithm 1 (lines 5 to 9) is responsible for creating virtual patients in the DTM corresponding to their real twins in the real hospital by taking

Algorithm 1 Initialization

```
Require: RealWorldClock,numberOfPatientPerActivity,j,i                              ▷ i, j array indices
1: procedure startInit(R,V)
2:     for i = 0 to R.length do                                     ▷ To cover all the activities in the real hospital
3:         j ← 0                                                     ▷ It will be used to loop through the IDs array
4:         numberOfPatientPerActivity ← (R.r[i]).numberOfPatients    ▷ Get the number of patients for each activity
5:         while j<numberOfPatientPerActivity do
6:             create ((V.v[i]).IDsV[j],(R.r[i]).IDsR[j])                        ▷ Create virtual patient in activity v[i]
7:             AssignNextDistPath(V.v[i]).IDsV[j])      ▷ Assign an expected next destination for the virtual patient
8:             j ← (j + 1)                                                   ▷ Update to read the ID of the next patient
9:         end while
10:        (V.v[i]).numberOfPatients ← (R.r[i]).numberOfPatients    ▷ Update the number of patients in activity v[i]
11:    end for
12:    DTMClock ← RealWorldClock                                 ▷ Assign the DTM clock to the real world clock
13: end procedure
```

into consideration their IDs and the current activities where they exist. Line 12 in the algorithm is responsible for initializing the DTM clock with the real-world clock. By using this algorithm, after pressing the **run button** on the simulation tool, the model will start in the same state as the real world.

By delving a little deeper into Algorithm 1, we can see that it is based on two parameters: the number of patients (line 4) and their locations (in which activity) (line 6). These parameters are appropriate for the DTM since this model is only used to monitor patients. When looking at the DTP, there is a third parameter that must be included at the initialization process: the amount of time the patient spends at each activity. In fact, predicting the patient's estimated time for beginning/completing the current activity is based on the time the patient spends in the activity before running the predictive model. For example, suppose the duration time for completing activity X is 5 minutes. Assume that before the predictive model is carried out, a patient called A has spent 3 minutes in activity X. To be more precise and to have a more accurate prediction, when the DTP is running, it must consider that the patient has already spent 3 minutes in activity X, and the time remaining for completing/finishing the activity is 2 minutes. At the initialization time of the predictive model, it must compute only 2 minutes for the duration time for activity X for patient A instead of 5 minutes. In this example, we suppose that the duration time for activity X is 5 minutes for the sake of simplicity, while in reality this time is a stochastic time. However, since the DTM is the main focus of this work, and the DTP is only used as a prototype to improve our future work, the time spent by the patient at each activity is beyond the scope of this work.

The question that arises in here is: "Due to using DES, is the data gathered from the real world going to be helpful in performing the initialization?". To answer to this question, a key point that must be understood first, which is that **collecting data depends on DES**. In other words, the data does not provide the accurate location of the patient because the collection of the data **happens only when the patient arrives at the sensor**. For example, when the patient arrives at the hospital entrance (the main door) where the sensor exists, the patient will be detected by this sensor. As a result, the data gathered by the sensor will mention that the patient is at the hospital entrance. If the patient continues to move to the waiting line, the sensor at the waiting line will detect the patient when he/she arrives. But in the path between the entrance and before arriving at the waiting line nothing is known about the location of the patient because our tracking for the patient depends on discrete events instead of tracking the patient continuously using an indoor real-time location system, among others.

Initializing the model based on discrete events alone is insufficient but could nevertheless be helpful because the stored information in the database represents only the most recent events of the patient and not the exact location of the patient. One of the possibilities for enriching the discrete events is that at the time we initialize the DT (DTM and DTP), a path for each created patient to follow will be proposed, as illustrated in Algorithm 1 (line 7). For example, if the database mentions that the patient is at the entrance, we create a virtual patient at the entrance and automatically this patient will move to the beginning of the waiting line. If the database mentions that the patient is at the entrance of the waiting room at the time of creating the patient, he will move to the chairs and so on. The argument for assigning a path for the virtual patient to follow is that the real patient always moves and does not remain at the location of the sensor. In other words, after creating the patient, he or she will be assigned a random path to follow in order to reflect reality and speculate on the patient's location by using discrete event systems.

The question that can be raised here is: "On what basis will the next destination of the virtual patient be selected? or, What are the characteristics of the selected path?" To answer these questions, we have established two rules:

1. If the most recent event concerning the patient mentions that the patient has arrived at the activity, then the selected path is to allow the patient to perform the activity. For example, if the patient arrives at the beginning of the waiting line, the chosen path is to allow the patient to enter the waiting line. If the patient arrives at the beginning of the waiting room, the selected path is to allow the patient to move to a chair.

2. If the most recent event indicates that the patient has completed the activity, then the historical data in the hospital information system will simply help to select the next path that the patient can follow. If the chosen path is incorrect due to faulty prediction, this problem will be resolved in the synchronization (Section 4.4). For example, if the most recent event mentions that the patient has completed his or her registration, thanks to the updated historical data, it is possible to predict that the patient's next step will be to go to the waiting room because the confidence interval of patients who finish their registration and go to the waiting room is 95%. If the waiting room is the wrong destination for this patient due to an incorrect prediction, the corrective step is synchronization. It will help to determine the correct destination for this patient based on the real-time events received from the real world. In this case, the synchronization step has been used not only for synchronizing purposes, but also to adjust the fault prediction that is based on historical data. However, synchronization will be discussed in Section 4.4.

Last but not least, one of the most important aspects to remember in both DTM and DTP is the clock. According to the findings in the literature review chapter, the DTM model must begin at the same time scale as the real world in order to represent reality. In the case of the monitoring model, the model's clock must be identical to the real-world clock during the initialization phase and must remain similar to the real-world clock until the DTM is turned off. For example, the ratio will be $\frac{RealWorldTime}{DTM\_Time} = 1$ at the time of starting the DTM until it is turned off. The real-world and simulation clocks must be identical at the beginning of running the predictive model (DTP), but when accelerating this model to predict the future, the simulation time scale will be greater than the time scale of the real-world clock. For example, the ratio will be $\frac{RealWorldTime}{DTP\_Time} < 1$. The following section will go through the synchronization process, which is the second phase after the initialization.

4.4 Synchronizing the DT Patient Pathways with the Real Patient Pathways

To demonstrate the concept behind synchronizing the DT patient pathways with the real patient pathways at each event detected in the real world, and to study the behavior of the DT in response to these events, a graphical model that is capable of providing rich visual information will be used. Due to using a DES to implement the proposed synchronization, a DES graphical implementation model is required. As a result, we need a model with specific rules that can handle the following requirements:

1. Elements to represent the patient activities (e.g., clinical, non-clinical, and administrative).
2. Elements to represent synchronization actions that will be used to inform the DT when an event occurs in the real world (e.g., when the sensor detects a real patient arriving at the hospital entrance, an action must be executed to inform the DT that there is a patient arriving at the hospital entrance according to the detected event, etc.).
3. Elements to represent the IoT devices that will be used to detect real patients (e.g., a sensor).
4. Elements to represent the events that were generated by the simulation tool in case the virtual patient arrives at an activity faster than the real patient (e.g., the virtual patient arrives in a waiting room while the real patient is still moving).
5. Elements to represent several patients in the pathway (two patients at the registration desk, one patient in the waiting room, etc.).
6. Illustrate multiple events occurring at the same time (two sensors at different activities detecting two different patients simultaneously).
7. A source element to represent when the patient enters the hospital.
8. A sink element to represent when the patient leaves the hospital.

To achieve the aforementioned requirements, two known models have been found, among others, that are suitable with some limitations: Petri Nets, specifically **object-oriented Petri Net**, and GRAFCET, specifically the version with this standard: **CEI IEC INTERNATIONAL STANDARD 60848** (commission et al., 2002).

The Petri Net was developed by Carl Adam Petri in his PhD thesis in 1962 (Petri, 1962). He intended to define a graphical and mathematical model for describing the relations between conditions and events that could be used in a variety of situations. There are two primary properties of the Petri Net (David et al., 1994) model that are worth mentioning. Firstly, it is possible to visualize behaviors like parallelism, concurrency, synchronization and resource sharing. Secondly, there are numerous theoretical methods for the analysis of these nets (Johnsson, 1999).

GRAFCET was proposed in France in 1977 as a formal specification and realization method for logical controllers (Johnsson, 1999). The acronym GRAFCET stands for ***Graphe Fonctionnel de Commande des Étapes et Transitions***. "It is a specification language for the functional description of the behaviour of the sequential part of a control system. It specifies the symbols and the rules for the graphical representation of this language, as well as for its interpretation. This standard has been prepared for automated production systems of industrial applications. However no particular area of application is excluded" (commission et al., 2002).

There are some commonalities between Petri Net and GRAFCET models, but there are also some distinctions. Table 4.1 summarizes our requirements and how they can be illustrated using these two models. The commonalities and distinctions between these two models are shown in the same table.

Based on the limitations shown in Table 4.1, there are two main differences between GRAFCET and Petri Nets: (1) Fireable transitions in GRAFCET fire simultaneously(Hrúz

Engineering Points of View		Models	
Req #	Requirements	Petri Net	GRAFCET
Req 1	How can the patient activities be represented?	Place	Continuous action
Req 2	How can the different types of actions performed in the virtual world be represented?	Place	Continuous action
Req 3	How can the synchronization action between the real world and the virtual world be represented?	Place	Stored action
Req 4	How can the IoT device be represented?	Transition	Transition
Req 5	How can the simulation tool events be represented?	Transition	Transition
Req 6	How can the patients be best represented?	Token	X
Req 7	How can the activation of simultaneous multiple events be depicted?	X	Simultaneous evolution rule
Req 8	How can the source component for generating a patient inside the model be represented?	Source transition	Source transition
Req 9	How can the sink component for removing a patient from the model be represented?	Sink transition	Pit transition

Table 4.1: Required elements for representing the synchronization approach in using Petri Net and GRAFCET

et al., 2007), while in the Petri Nets, fireable transitions can fire only one at a time (Hrúz et al., 2007; Van Der Aalst, 1998). (2) The concept of a “token" is not included in GRAFCET; a “mark" shows if a step is active or not (commission et al., 2002). In Petri Net, the concept of a token is included, and there is a possibility for a step to have multiple tokens (Hrúz et al., 2007).

Based on the previous comparison, a Petri Net or a GRAFCET alone will not be sufficient for meeting our requirements. As a result, we have chosen to design a new graphical chart by combining the logic from both of the illustrated models. This chart will help us attain our goal, which is referred to as the **GRAFNet (GRAFCET Petri Net)**. We shall inherit the token concept from the object-oriented Petri Net in this chart. In reality, in an object-oriented Petri Net, each step can have numerous tokens, and these tokens can be distinguished by the various properties that are associated with each token (Miyamoto et al., 2005). The GRAFCET simultaneous evolution rule will be used to inherit the concept of simultaneous activation for distinct events. We will utilize the same GRAFCET logic for the source and sink components. Furthermore, the structure and the rules of GRAFCET will be dominant in our tool because GRAFCET is easier for the developer to understand and it has a few more specific properties compared to the Petri Net. Different details about GRAFNet, with the meaning of each element, will be clarified in Section 4.4.1.

4.4.1 GRAFNet : A customized graphical chart for illustrating our synchronization approach

GRAFNet includes tokens, steps associated with actions, transitions associated with receptivities, and directed links connecting steps and transitions as follows:

- Token: In this work, a token corresponds to a patient, where each token is considered an object. Each object has at least one attribute. For example, Patient Identification (ID) can be an attribute for the patient token. This attribute will be used to distinguish the tokens from each other. More than one token can exist in the same GRAFNet chart.

- Steps associated with actions: In this work, a step will be used for defining the state of a part of the patient pathway. It can have zero or more tokens, and it is considered active if it includes at least one token. Each step is associated with zero or more actions. The zero action step is called an empty step, as shown in Figure 4.1 (a), whereas a non-empty step (step with action) is depicted in Figure 4.1 (b). The red circle in Figure 4.1 (b) corresponds to a token.

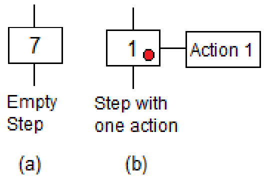


Figure 4.1: Step with/without action

Step i is represented by a numerically identified square. It has a state variable known as step variable Xi. This variable is a Boolean variable that has a value of 1 if the step is active and 0 otherwise.

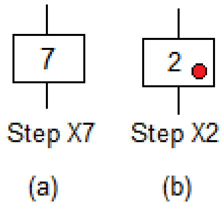


Figure 4.2: GRAFNet step

There are two classes of actions which are defined in the GRAFNet chart. The first class represents a patient activity (e.g., waiting line, waiting room, registration desk, etc.). This class is represented by continuous action. The second class is used to notify the DT when an event occurs in the real world (for example, the patient arrives at the registration desk, the patient leaves the hospital, etc). This class is represented by stored action.

The stored action will be used to maintain a synchronization point between the virtual and real worlds, whereas the continuous action will be used to keep the DT activity running until a real world event indicates that the activity is finished. Moreover, continuous action will be used to represent the action executed in the DT. Furthermore, with GRAFNet, both types of actions are considered objects. As previously stated, each object can have specific attributes that distinguish it from other actions such as name, ID, etc. Figure 4.3 demonstrates the continuous and stored actions.

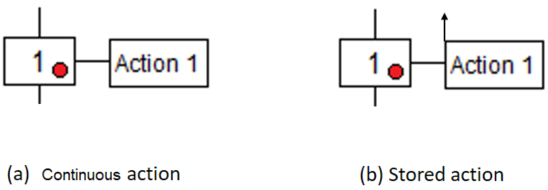


Figure 4.3: Continuous action and stored action

- Transitions associated with receptivities:

  A transition denotes the possibility of evolution between two steps. Each transition is associated with a logical condition called receptivity, (R), as shown in Figure 4.4, which expresses the condition required for progressing from one step to the next. In other words, transition can be thought of as a door; if this door is open, a particular token can move from step one to step two. However, if the door is closed, the token will be unable to move and will remain at step one. The state of the door is determined by the value of R, which is determined by input information from various sources such as sensors, operators, timers, counters, and so on. If the value of R is one, it means the condition is true and the door is open, and the token can move from one step to the next. However, if the value of R is zero, it means the condition is false and the door is closed, and the token cannot move.

  In this work, the transitions will be used on the basis of the following assumptions:

  1. The transitions in GRAFNet real patient pathway represent sensors (discrete events), except if the receptivity R is explicitly defined to be equivalent to 1 or depends on duration, counters, etc.
  2. The receptivity R, associated with the transitions in the GRAFNet virtual patient pathway, is dependent on the input information, which can come from the stored actions in the GRAFNet real patient pathway or the simulation tool events, and some values of R can be explicitly defined to be equivalent to 1, or depend on duration, counter, etc.

The GRAFNet chart in Figure 4.4 illustrates the transition as a sensor. At the time the sensor detects an event from the real world, the stored action 1 associated with step 1 will be executed to inform the GRAFNet virtual world that an event has happened in the real world. Thus, the value of R in the GRAFNet virtual world will be changed to true to indicate that an event from the real world has been received.

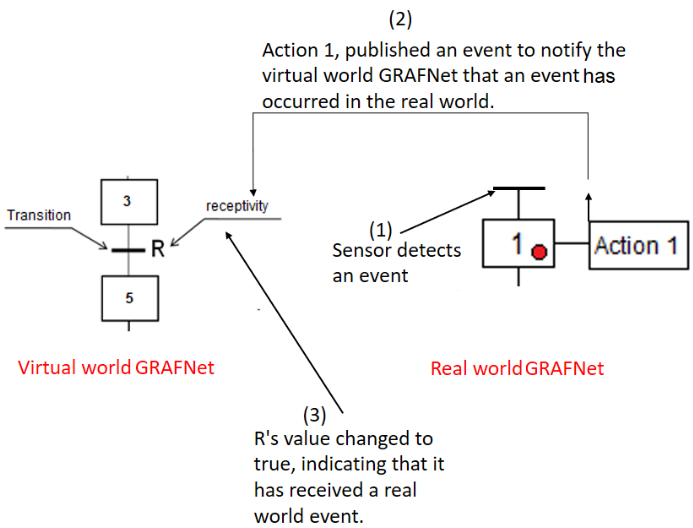


Figure 4.4: Transition

It is worth noting that the transition can be treated as an object. For example, suppose we have various types of tokens in the GRAFNet chart. A token, for example, can represent a doctor or a patient, among other things. In this case, each transition must have a receptivity for each type of token. In this work, the token only represents the patient.

- Directed links connecting steps and transitions:

  These are lines that connect one or several steps to a transition, or a transition to one or several steps, as shown in Figure 4.5.

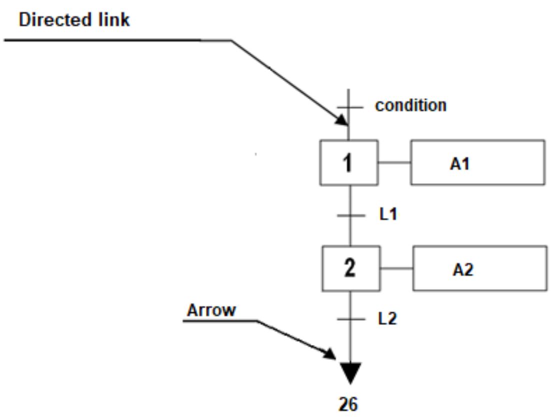


Figure 4.5: Directed Links

The arrow in Figure 4.5 is called a jump-to-label symbol (linked label). To avoid cluttering up the diagram, links can be replaced with a jump to a label. The jump to the label works in the same way as any other directed link, thus this is simply a decorative symbol. However, the value of this label can be the number of the destination step. For example, step 26. If step 26 exists on a different page, there is also the possibility of mentioning the page number. For example, step 26 page 3.

- Source transition: this is a transition which does not have any preceding steps, as illustrated in Figure 4.6. By convention, the source transition is always enabled and it is fired as soon as its transition-condition (receptivity) is true. In this work, the transition condition is considered to be true when the sensor detects an event.

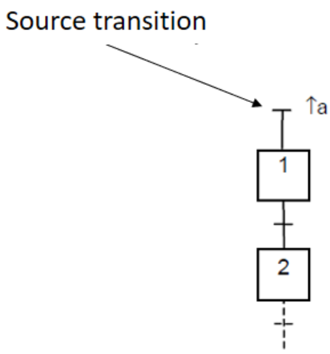


Figure 4.6: Source transition

Due to the fact that numerous patients can exist in the patient pathway, the source transition will be used to generate several tokens, where each token is a reference for each patient. For example, if there are three patients in the patient pathway, this means the GRAFNet chart will include three tokens, each one referring to a patient.

- Sink transition: this is a transition which has no succeeding step. When the sink transition is enabled and when its associated receptivity is true (R=1), the token will be removed from the GRAFNet chart.

  Normally, the patient will be discharged from the hospital following recovery/treatment. To illustrate this case, the sink transition will be used. For example, when the sensor at the exit detects a patient leaving the hospital, the sink transition will be activated and the token referring to the detected patient will be removed from the GRAFNet chart. Figure 4.7 illustrates the sink transition element.

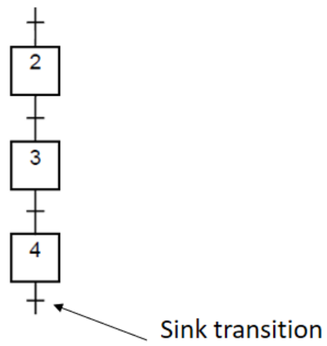


Figure 4.7: Sink transition

As with the other GRAFNet elements, source and sink transitions can be considered objects that generate and remove different types of tokens. The source transition can generate a token corresponding to patients, doctors, and other resources, whereas the sink transition can be used to remove these tokens. However, representing the resources using GRAFNet is outside the scope of this research work.

- Rising edge of a logical variable: the symbol ($\uparrow$) means that the receptivity (R) is only true for the state of variable changes (rising edge: changing from value 0 to value 1), as depicted in Figure 4.8.

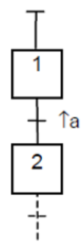


Figure 4.8: Rising edge

- Usual simplification (t1/X*) : this delays the step X* by a time t1; then, the transition condition (receptivity) becomes false on deactivation of the step X* that activated the delay. See Figure 4.9.

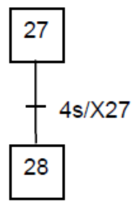


Figure 4.9: Usual simplification (Delay)

- Parallel bars: The beginning and the termination of a parallel branch are indicated by a parallel bar. A parallel bar of type and-divergence (parallel split) is used to mark the start of such a branch, while a parallel bar of type and-convergence (parallel join) is used to indicate the end, as illustrated in Figure 4.10.

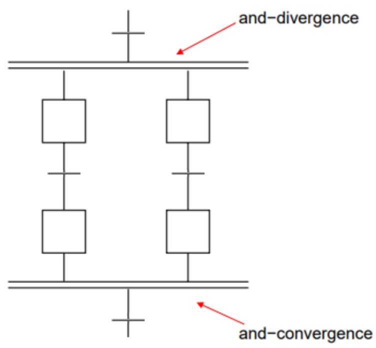


Figure 4.10: A parallel branch

To build our GRAFNet, we have mentioned that the most important elements that we needed to use were from GRAFCET and Petri Net. Other elements that have not been mentioned can be included without affecting the five evolution rules that will be illustrated in this section.

The comprehensive behavior of any GRAFNet chart can be obtained by applying the following five evolution rules:

Rule 1 Creating tokens:

At any time, a source transition is always enabled, and it is fired as soon as its receptivity changes to true. Firing a source transition consists in adding a token to each of the output steps of this transition.

Rule 2 Activation/Deactivation of a step:

A step is considered to be active if it has at least one token, otherwise it is not active.

Rule 3 Firing of a transition:

A transition can only be fired if all of the steps immediately preceding it are active. The transition is then said to be enabled. A transition is fireable only if it is enabled and when the associated receptivity is true. A fireable transition must be fired immediately. Firing a transition consists in withdrawing a token (depending on output rules: First In, First Out (FIFO), Last In, First Out (LIFO), etc.) from each of the input steps of this transition and in adding a token to each of the output steps of this transition.

Rule 4 Evolution of the active steps:

The firing of a transition leads to the activation of all the following steps and the deactivation of the previous steps if the number of tokens is null.

Rule 5 Simultaneous fireable transitions:

When multiple transitions are simultaneously fireable, they are simultaneously fired.

Related to the above-mentioned GRAFNet rules, Table 4.2 summarizes the technique of origin for each of these rules. For example, the table mentions the original source for each of these rules, as well as whether our research work added updates to the original rule or not.

#	GRAFNet rules	Technique of origin	Update on the original rule
1	Creating tokens rule	GRAFCET and Petri Net	Proposed by this research work
2	Activation/Deactivation of a step	GRAFCET and Petri Net	Proposed by this research work
3	Firing of a transition	GRAFCET	Different output rules could be added: 1. Fire token based on FIFO 2. Fire token based on LIFO 3. Fire token based on condition 4. etc.
4	Evolution of the active steps	GRAFCET	Withdrawing a token from the previous step does not mean changing its state to inactive.
5	Simultaneous fireable transitions	GRAFCET	No updates

Table 4.2: The original sources for GRAFNet rules

4.4.2 Highlighting synchronization issues in discrete event simulation

This section aims to address the question of how to synchronize the DTM with the real-world patient pathways. Behind this question, there are four critical issues for consideration:

- How can the synchronization between the virtual patient and the real patient be done based on discrete events when taking into account that discrete events do not track the patient continuously at each point of time?
- Is the clock considered part of the synchronization or just related to the initialization?
- Which model of DT must be synchronized with the real world: the DTM or the DTP?
- How is it possible to benefit from the DTM in order to initialize the DTP? (Why is the DTM important?)

In this work, we consider synchronization to be the second phase after initialization, in which the virtual patient pathways must be in the same state as the real patient pathways at each event detected in the real world. If an event occurs in the real world, the same event must occur in the DT at the same time or in near real time. Thus, the DT must follow the dynamic behavior of the real world to be in the same state.

Using the previously defined sets (R and V) in Section 4.3, set V must have the same values as set R at each detected event. When the real patient arrives at the hospital entrance, set R will be updated to include the location and the ID of this patient. In near real time, a virtual patient corresponding to the real one must be created at the entrance door of the DTM. In this case, set V will be updated to be identical to set R. The problem here is that the two sets, R and V, include only the patient's location corresponding to the most recent event detected by the sensor. For example, the two sets will show the patient at the entrance. But in fact, the patient is not at the entrance because the patient in the real world is continuously moving. For example, when the real patient arrives at the entrance, the following procedures will be followed: (1) the entrance sensor detects the real patient, (2) the sensor publishes an event to the DTM in the virtual world, (3) the real patient moves to the next destination, and (4) at the same time, a virtual patient will be created at the virtual hospital's entrance based on the received event. Imagine that in step (3) that the real patient has not yet arrived at his/her next destination. This means that the virtual patient will stay at the virtual entrance until he/she receives the next event from the real patient to decide his/her next destination. The reason behind this is related to using discrete events for tracking the real patient instead of using continuous monitoring such as the Indoor Real-Time Location System (RTLS). Imagine that there are many patients moving from the hospital entrance to the next destination, where they haven't arrived yet. This will lead to more than one patient being stuck at the virtual hospital entrance, which does not reflect the current behavior of the real hospital where the real patients are moving while the virtual patients are stuck/blocked waiting for the next event, as illustrated in Figure 4.11.

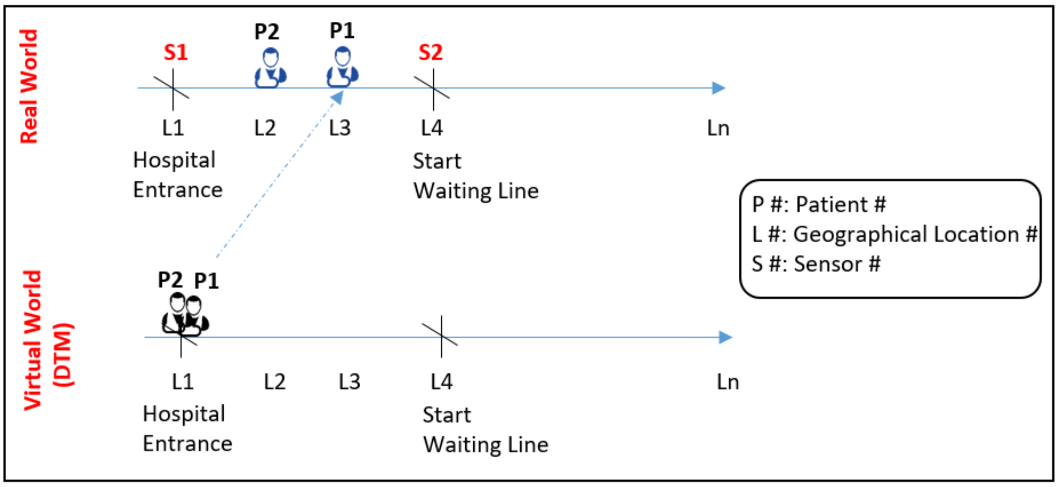


Figure 4.11: Synchronization issue

In Figure 4.11, the real patients P1 and P2 are at locations L3 and L2 respectively, while the virtual patients corresponding to P1 and P2 are stuck at the hospital entrance (L1) awaiting the arrival of the real patients at sensor S2. According to what is shown in this figure, there is a compatibility problem in which the virtual world (DTM) does not accurately reflect the actual behavior of the real world. The aim of the synchronization in this work is to get the DTM as close to the real world as possible. That is to say, if the real patient moves to the real waiting line after the entry, we want the virtual patient to move to the virtual waiting line as well. This is one of the main objectives of this work; having the current state of patient pathways in the DTM, then using the current state from the DTM to feed the DTP to predict the near future. We believe that predicting the future starting from the current state will be more accurate for predictive analysis.

To address the synchronization issue depicted in Figure 4.11, the created virtual patient will be assigned a dynamic random destination/path to follow after creation. This random path will be generated using dynamically updated historical data stored in the hospital information system's database (HIS). The same issue may arise when the real patient completes the current activity and wishes to move on to the next activity. The virtual patient will be held up until he/she receives an event informing him/her of the next destination. This issue can be solved by repeating the process used to create the virtual patient. In other words, when the real patient completes an activity, the virtual patient is released from the same activity by assigning him/her a path to follow. Finally, when the real patient leaves the hospital, the virtual patient must be removed from the DTM.

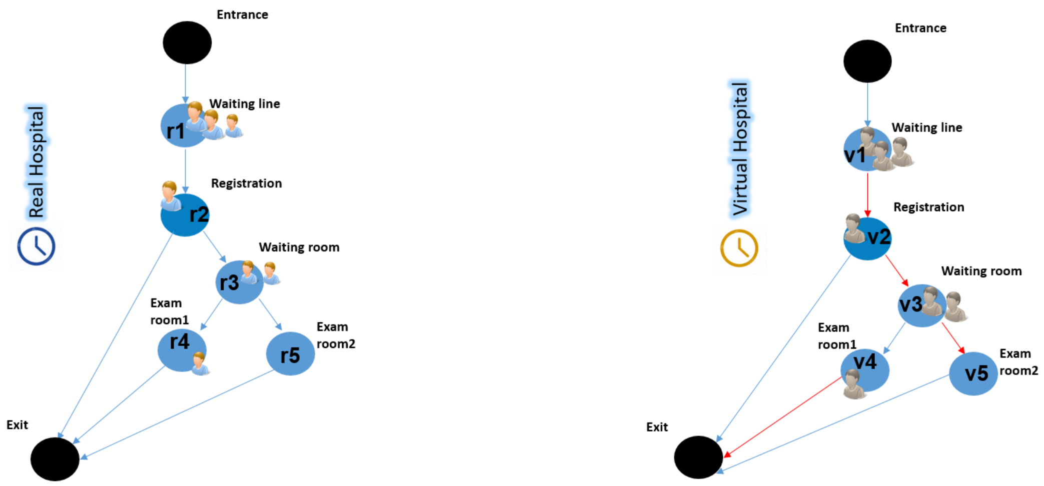


Figure 4.12: A type of synchronization between the real patient pathways and virtual patient pathways in the hospital

Figure 4.12 illustrates a type of synchronization between the real patient pathways and the virtual patient pathways in the hospital. Here, there are five activities for both hospitals (r1,v1: waiting line, r2,v2: registration, r3,v3: waiting room, r4,v4:exam room 1, r5,v5: exam room 2) where each activity has the same number of patients (r1,v1: 3 patients, r2,v2: 1 patient, r3,v3: 2 patients, r4,v4: 1 patients, r5,v5: 0 patients). Entrance, exit, and clock also appear in this figure due to their importance in synchronization. The synchronization algorithms behind this figure will be illustrated in this section. Due to the fact that there are differences in walking speeds between the real and the virtual patients, along with the different directions that the two patients may take, three outstanding synchronization issues must be resolved (Karakra et al., 2020):

- Real patient (RP) is faster than virtual patient (VP). For example, Speed (RP2,VP2)> 0, as illustrated in Figure 4.13.
- Real patient is slower than virtual patient. For example, Speed (RP1,VP1)< 0, as illustrated in Figure 4.13.
- Wrong destination/path: the virtual patient does not know the next destination until the real patient actually arrives. For example, the virtual patient goes to activity X (wrong activity), while the real patient goes to activity Y. For example, Different-Direction (RP3,VP3), as illustrated in Figure 4.13.

Algorithm 2 shows the process of synchronizing virtual patient pathways with real patient pathways, taking into account the illustrated synchronization issues. This depicts a high-level overview of the main functions that must be used. These functions may be implemented

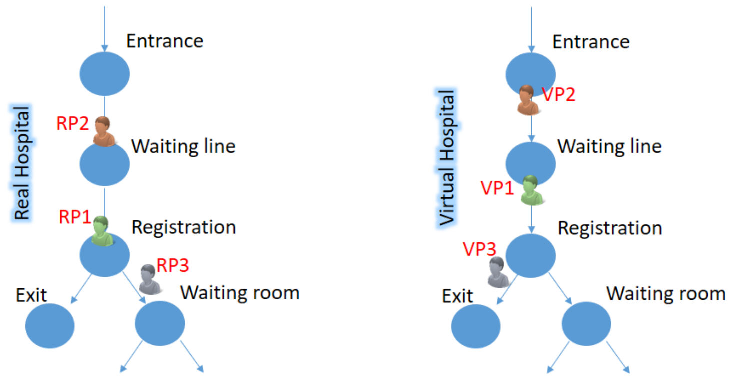


Figure 4.13: Three synchronization issues

differently by each programmer. More clarifications for these functions will be discussed using the GRAFNet chart in Section 4.4.3. However, in this algorithm, various shortcuts have been used, such as: VP stands for virtual patient, CreateVP stands for create virtual patient, RemoveVP stands for remove the virtual patient, and SynchVPLocation means synchronize the virtual patient's location with the real patient's location.

Algorithm 2 Synchronization (Karakra et al., 2020)

Require: *EventType* ▷ Enter, Exit, or Move

```
procedure StartSynch()
    EventType ← waitRealWorldEvent()
    if EventType= Enter then
        VP ← CreateVP(HospitalEntrance, RP_ID)
        AssignNextDistPath(VP)
        ReleaseToMove(VP)
    else if EventType= Exit then
        RemoveVP (VP)
    else
        SynchVPLocation (VP,RP)
    end if
end procedure
```

For the **SynchVPLocation** procedure, there are two approaches that can be used to perform the synchronization: (1) removing and creating the patient approach, and (2) accelerating the patient approach. In the first approach, the real patient arrives at activity X while the virtual patient is still walking to the same activity. In this case, the virtual patient must be removed from the model and recreated with the same ID at activity X. This scenario depicts the situation in which the real patient is faster than the virtual patient.

If the virtual patient arrives at activity X while the real patient is still walking, the virtual patient must be blocked from starting the activity until the real patient arrives at the same activity. This case represents the scenario in which the real patient is slower than the virtual patient. The third scenario occurs when the real patient and the virtual patient have gone to two different activities. In this case, the virtual patient must be released to the same location as the real patient. This can be solved by removing the virtual patient from the model and creating him/her in the same location as the real patient.

Algorithm 3 Update the location of the virtual patient

```
Require: EventType
 1: procedure SynchVPLocation(VP,RP)
 2:     if RP faster than VP then
 3:         Accelerate (VP)
 4:         PerformAtivity()
 5:     else                                         ▷ VP faster than RP
 6:         Block (VP)
 7:         EventType ← simulationToolEvent()
 8:         if EventType= SameActivity then
 9:           PerformAtivity()
10:         else                                     ▷ DiffActivity
11:             Accelerate (VP)
12:             PerformAtivity()
13:         end if
14:     end if
15:      Update IDs                                  ▷ Update globally unique IDs
16: end procedure
```

The problem in "removing and creating the patient approach" is that removing and recreating the patient results in the loss of all statistics pertaining to this patient, such as the different activities the patient has visited, the different duration times for each activity, travel distances, etc. In addition, the model may crash as a result of numerous requests for the removal and creation procedures for the same patient due to many synchronization events. As a result, in this work, we recommend using the "accelerating the patient approach" to preserve the statistics of each patient, as it will not be necessary to copy the statistics from the removed patient and paste them again in the recreated patient. This will help reduce the time-consuming copy/paste procedures and the large memory allocation for these procedures, as well as preventing the model from crashing.

In "accelerating the patient approach", if the real patient is faster than the virtual patient, the virtual patient must be accelerated to the same location as the real patient, as illustrated in Algorithm 3 (line 3). If the virtual patient is faster, the virtual patient has to be blocked from starting the activity until the real patient arrives at the same activity, as depicted in Algorithm 3 (line 5). In the case of a different direction, the virtual patient must be released to the same location as the real patient, as shown in Algorithm 3 (lines 11 and 12). This can be resolved by accelerating the virtual patient to the same location as the real patient.

The question now is: "Is the clock considered a part of the synchronization or just related to the initialization?" The answer here depends on the type of model of DT. For example, if the DT is used to monitor the patient pathways, the clock must be a part of the synchronization. In other words, the time scale of the DT clock must be equivalent to the time scale of the real-world wall clock to study the behavior of the two worlds at the same time. Hence, to do the synchronization, (1) the DT will be initialized with the current state of the real world, then (2) the simulation clock must be set to the same time as the real-world wall clock; then, (3) the DT will start running.

If the DT is used for predicting the future, the clock should be a part of the initialization only at the initialization step. In fact, in this work, we are trying to predict the future while starting from the current time. After that, the predicting model must be speeded up. As a result, the time scale of the virtual world clock is greater than the time scale of the real world; for example, x:y (x seconds in the real world is simulated in y seconds in the virtual world, where x>y).

Now, we return to the question: "How is it possible to benefit from the DTM to initialize the DTP? (Why is the DTM important?)"

The answer to this question is that we believe that taking a clone from the current hospital as an input to the predictive model and then running the predictive model to anticipate the future based on this clone will lead to more accurate prediction. For this reason, a monitoring model is seems to be a good choice for this requirement.

To simplify the synchronization mechanism between the virtual patient pathways and the real patient pathways, Section 4.4.3 illustrates the different concepts behind this synchronization. This section shows the different steps for the real patient pathway life cycle from entrance to discharge and illustrates the corresponding steps in the virtual patient pathways. Two GRAFNets have been developed for the real and the virtual patient pathways. The different events and the synchronization points are discussed in this section.

4.4.3 Using the GRAFNet chart to demonstrate the concepts behind synchronization

This section depicts a generic GRAFNet that represents real world patient pathways, virtual patient pathways, and their synchronization. The aim of this section is to answer the following questions through the GRAFNet graphical chart :

- What are the different concepts behind synchronizing the virtual patient pathways and the real patient pathways?
- What are the various choices and measures that may be taken in the real patient pathways and virtual patient pathways on a patient journey?
- Because the DT must be as close as possible to the real world, does this that mean the real patient pathways and the virtual patient pathways will have the same GRAFNet chart in terms of steps, transitions, actions, etc.?

In order to answer the questions mentioned above, this section will explain each GRAFNet separately (a GRAFNet for the real patient pathways, and a GRAFNet for the virtual patient pathways), and then show their synchronization.

Figure 4.14 illustrates the GRAFNet for the real patient pathways, where the steps in this GRAFNet are associated with real-world actions. Two types of action can be distinguished in this GRAFNet: the first type, which is responsible for generating events, is represented by stored actions. The second type, which represents real-world activities (e.g., waiting line, exam room, etc), is represented by continuous actions. The transitions in this GRAFNet represent sensors in the real world, except for the transitions with R=1. R=1 means the receptivity associated with this transition is always true: as soon as there is a token in the immediately preceding step (the transition's upstream steps), this transition will be fireable. This GRAFNet executes as follows:

1. In the beginning, the source transition RP_arrive_Entrance of this GRAFNet is always enabled (by convention).
2. As shown in Figure 4.15 (b), when the transition RP_arrive_Entrance detects a patient arriving at the hospital entrance, the condition associated with this transition will be changed to true (the value of transition RP_arrive_Entrance is changed to 1). This transition is considered to be fireable.

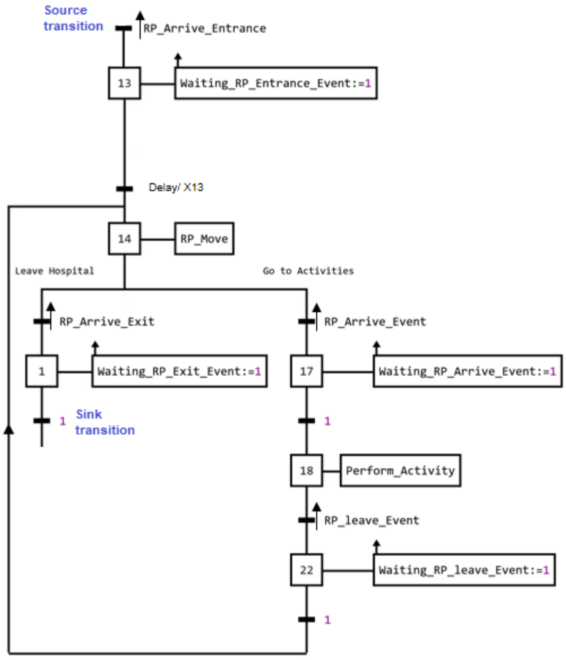


Figure 4.14: Generic GRAFNet for the real patient pathways, see Figure 6.2

3. A token will be created at step 13, indicating that there is a real patient arriving at the hospital entrance, as demonstrated in Figure 4.15 (c). The state of step 13 will change to active (has at least one token). Due to the activation of step 13, the action associated with this step will be executed to inform the DT (in this case the virtual patient pathways) that a new patient has arrived at the hospital entrance.

4. The token that is in step 13 will move to step 14 to activate it, as shown in Figure 4.16. As a result, step 13 is deactivated (step is empty or has no token). The action associated with step 14 will be executed to illustrate the moves of the patients in the real patient pathways.

5. As illustrated in the GRAFNet in Figure 4.16, following step 14, the patient has two decisions to make. The first one is that the patient can decide to go home. In this case, the left branch after step 14 will be executed. Or the patient can decide to go to an activity, and in this case the right branch will be executed. Suppose the decision of the patient is to go to the first activity after entering the hospital:

6. When the transition RP_Arrive_Event detects a patient arriving at the activity, the condition associated with this transition is changed to true, as shown in Figure 4.17 (a), and the token moves from step 14 to step 17, as shown in the same figure part (b). As a result, step 14 is deactivated (no token) and step 17 is activated. The action associated with the step 17 will be carried out to notify the DT that the patient has arrived at activity X.

7. The receptivity of the transition that appears in Figure 4.17 (c) is equivalent to 1. This means the condition is always true. Thus, the token will move to step 18 immediately. This leads to deactivating step 17 and activating step 18. The action that is associated with step 18 will be executed to illustrate that the patient is inside the activity.

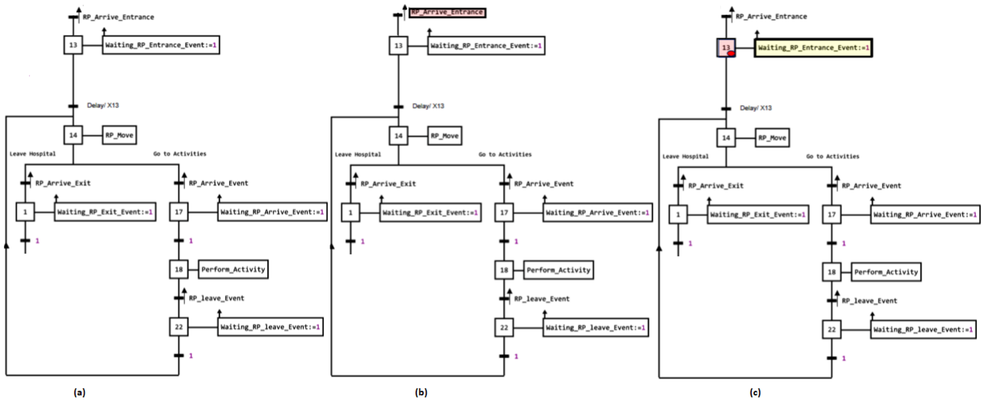


Figure 4.15: Sample execution for the generic GRAFNet real patient pathway: start executing the GRAFNet, see Figure 6.3

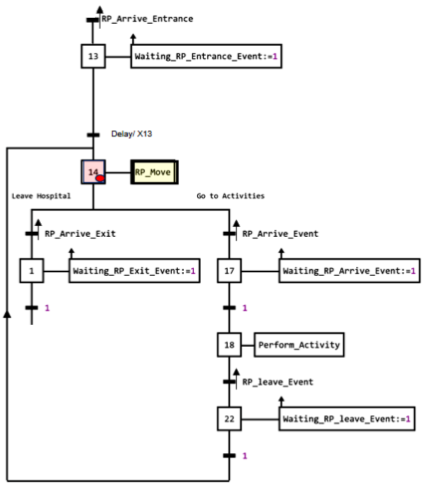


Figure 4.16: Sample execution for the generic GRAFNet real patient pathway: execute moving activity, see Figure 6.4

8. When the transition RP_leave_Event detects a patient finishing the activity, the condition associated with this transition is changed to true, as shown in Figure 4.17 (d). As a result, the token will move from step 18 to step 22, as demonstrated in Figure 4.17 (e). This leads to deactivating step 18 and activating step 22. The action associated with step 22 will be executed to inform the DT that the patient has finished the activity.

9. Last but not least, the token will return to step 14. This leads to deactivating step 22 and activating step 14 as in the previous Figure 4.16. The action associated with step 14 will be executed to illustrate the moving process of the patient. After that, the patient can decide to go to the next activity or to leave the hospital.

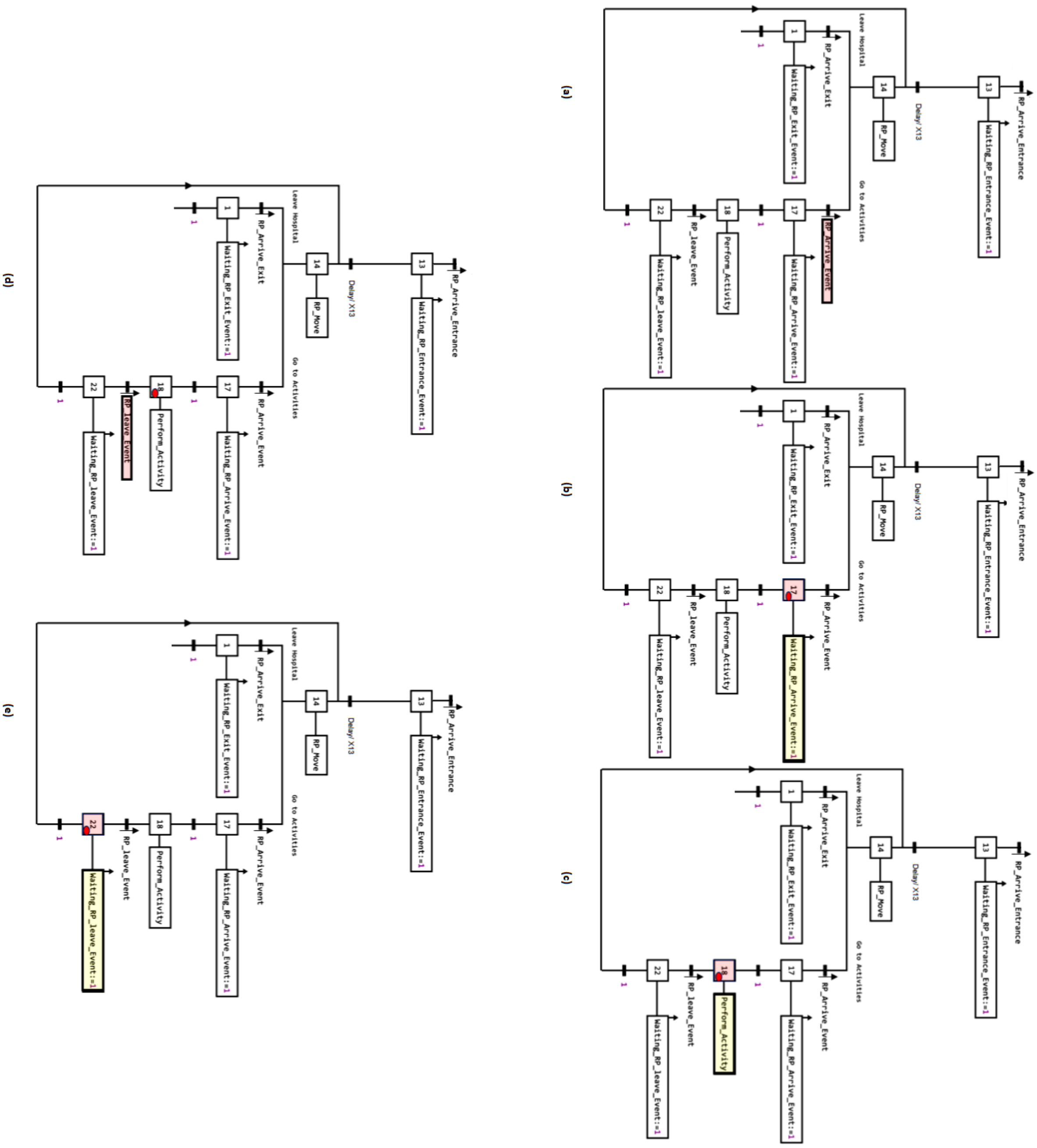


Figure 4.17: Sample execution for the generic GRAFNet real patient pathway: perform activity, see Figure 6.5

10. To change the scenario, assume that the patient wants to leave the hospital. In this case, when the transition RP_Arrive_Exit detects the patient arriving at the hospital exit, as shown in Figure 4.18 (a), the token will move from step 14 to step 1. As a result, step 14 will be deactivated and step 1 will be activated, as demonstrated in Figure 4.18 (b). The action associated with step 1 will be executed to inform the DT that the patient has arrived at the hospital exit. Because the receptivity of the transition after step 1 is equivalent to 1, the token will move to the sink step which is responsible for removing the token from the GRAFNet because the real patient corresponding to this token has left the hospital.

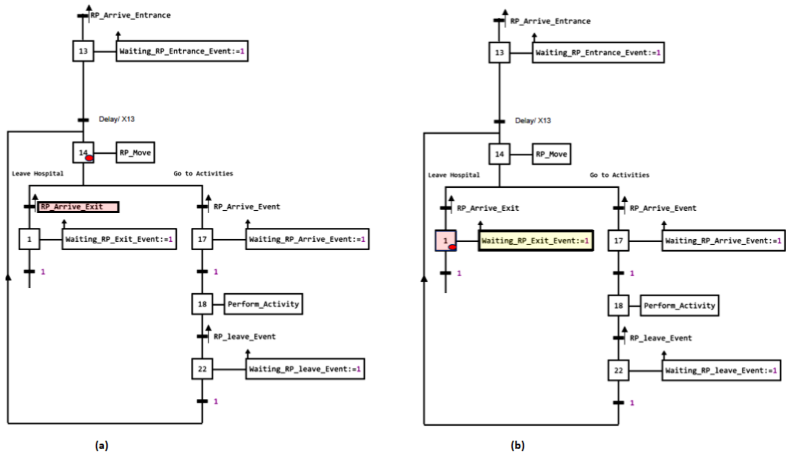


Figure 4.18: Sample execution for the generic GRAFNet real patient pathways: leave the hospital, see Figure 6.6

> It is worth mentioning here that when the token moves from step X1 to step X2, this will lead to activating step X2 and deactivating step X1. The reason behind this is because we have no token in step X1, as illustrated in Figure 4.19 (a). For example, if there are two tokens in step X1, and one of them moves to X2, step X1 will still be active, and the value of R will determine whether the remaining token in step X1 is fired to step X2, as demonstrated in Figure 4.19 (b). The second token will be fired to step X2 when R becomes true once more. As soon as there are no more tokens in step X1, step 1 will become inactive. This feature is a part of GRAFNet and not GRAFCET.
>
> One of the key reasons for having this feature is because the action associated with step X1 could depict a hospital waiting room with multiple patients waiting for the second activity. Only one patient is allowed to go on to the next activity at a time.

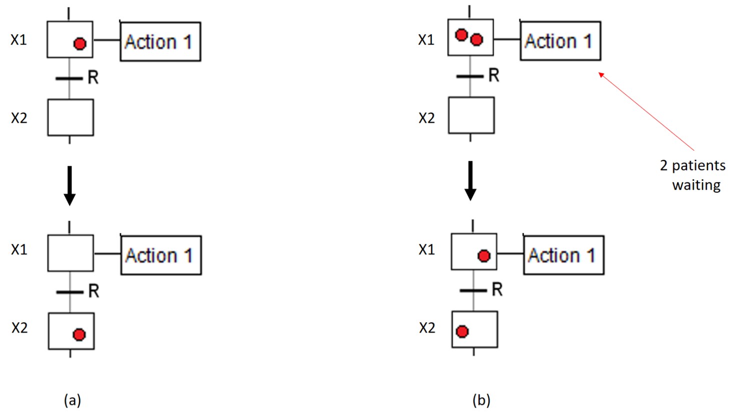


Figure 4.19: Activate and deactivate the GRAFNet steps

The presented GRAFNet illustrates an example of patient behavior in the real patient pathways. It details the different steps, transitions and synchronization actions. It shows that the stored actions have been used to inform the DT about the locations of the real patients, whereas the continuous actions have been used to represent the activity the patient undergoes or does. In this GRAFNet, there are multiple transitions with different kinds of receptivities. For example, the value of some receptivities depends on whether an event happened in the real world or not. Other receptivities store duration time (Delay/X13) to illustrate that a delay can exist between the different steps. The last type of receptivity in this GRAFNet, store the value 1, indicates that the token can move from one step to another without needing to wait for an event to happen. The transition that has this receptivity is used as an auxiliary transition to facilitate the moving of the token to the next step without evaluating any condition.

Finally, in the discussed GRAFNet, a small rising edge can sometimes be seen, attached to the right side of a transition (↑). We have used this edge to indicate that the detection of an event will happen only when the value of receptivity is 0 (false condition). In other words, before detecting an event, if the value of the receptivity is 0, this means this receptivity is ready to receive an event from the real world. After receiving an event, the value of the receptivity will change to 1 (true condition). This means that the receptivity cannot change its state until its value returns to 0. So, we have used this edge to indicate that there is a short delay between the first event and the second event. This delay depends on the time needed for the receptivity to change its value from 1 to 0. This logic is also applied to the real world sensor. For example, if there is a very short delay between two events, there will be a short delay between detecting two patients by the same sensor.

Figure 4.20 depicts a generic GRAFNet for the virtual patient pathways. The steps in this GRAFNet are associated with the virtual world's actions (the scripts that will be executed in the DT e.g., create patient, remove patient, etc.). These actions will be carried out as a result of the events that the DT may receive from the real patient pathways (called external events) or from the simulation model (called internal events). The external events in this GRAFNet are represented by the data coming from stored actions. In this GRAFNet, if the real patient arrives at the activity faster than the virtual patient, the actions that will be taken will be determined by external events; however, if the virtual patients arrive first, the actions will be determined by internal events.

When the DT (virtual patient pathways) is started, the source transition (Waiting_RP_Entrance_Event) is always enabled (by convention), and waits for an event from the real patient pathways. When the real patient arrives at the entrance door in the real hospital, the following procedures will be carried out in the GRAFNet virtual world and in the following order:

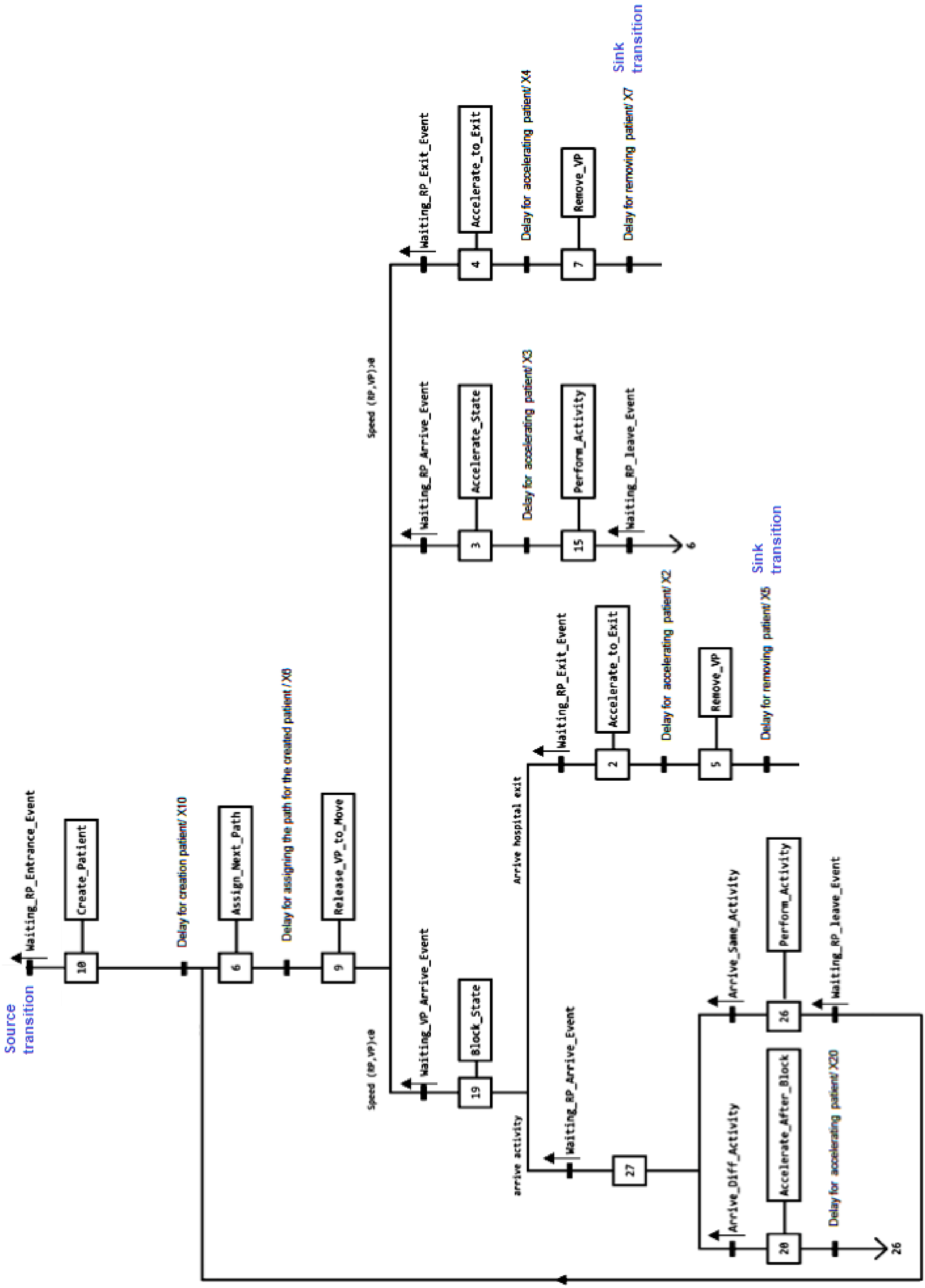


Figure 4.20: Generic GRAFNet for the virtual patient pathways, see Figure 6.7

1. The condition associated with the transition Waiting_RP_Entrance_Event will change to true. This means it will be fireable. A token will be created at step 10. Thus, the state of step 10 will change to active. Due to the activation of step 10, the action associated with this step will be executed to create a virtual patient at the entrance of the virtual hospital, corresponding to the real patient who generated the event. However, the token now existing in step 10 will be considered the representation of the virtual patient that was created due to the execution of the create patient action. In fact, the process of creating this patient will take a very short period. After that, the token will move from step 10 to step 6 to assign a random path for the created patient to follow. In this case, step 10 will be deactivated and step 6 will be activated for a short delay. Then the created patient will be released to follow the proposed path when the token moves from step 6 to step 9. As a result, step 6 will be deactivated (no tokens there) and step 9 will be activated, and the patient will continue following the proposed path until an event comes from the real patient pathway that mentions that a real patient is arriving at an activity or exit door faster than the virtual patient (right branch) or an event comes from the simulation tool that mentions that the virtual patient arrived first at the next location (left branch).

2. Suppose that the virtual patient arrives first at the activity, while the real patient is still walking. This means the condition associated with the transition Waiting_VP_Arrive_Event will change to true. Thus, the token passes from step 9 to step 19. This will deactivate step 9 and activate step 19. As a result, the action associated with step 19 will be performed to demonstrate the process of preventing/blocking the virtual patient from beginning the activity until an event from the real patient pathway indicates that a real patient has arrived at the activity (left branch) or the exit door (right branch).

3. Assume that the event from the real patient pathway mentions that the real patient has arrived at the activity. As a result, the condition associated with the transition Waiting_RP_Arrive_Event will change to true. In this case, from the point of view of the virtual patient pathway, the real patient has two options: either arrive at the same activity where the virtual patient is blocked (right branch), or arrive at a different activity (left branch).

4. Assume that the real patient arrives at the different activity. This means that the real patient arrives at an activity that is different from the activity where the virtual patient is blocked. In this case, the condition associated with the transition Arrive_Diff_Activity will change to true. As a result, the token will move from step 27 to step 20. This will deactivate step 27 and activate step 20. The action associated with step 20 will be executed to accelerate the virtual patient to the same location as the real patient. For example, if the virtual patient is blocked at activity X, and the real patient arrives at activity Y, the virtual patient must be accelerated to activity Y. This acceleration may create a very short delay. Then, the token will move from step 20 to step 26. This will deactivate step 20 and activate step 26. After that, the action associated with step 26 will be executed to illustrate that the virtual patient is performing the same activity as the real patient. The virtual patient will continue performing the activity until a new event comes from the real patient pathway that mentions the real patient finishing the activity. When this event is received, in this case, the condition associated with the transition Waiting_RP_leave_Event will change to true. As a result, the token will move from step 26 to step 6 to repeat the same process of assigning a path for the virtual patient to follow, and then release this patient to move. Last but not least, step 9 will be activated until an event comes from the real patient pathway that mentions a real patient arriving at the activity or exit door faster than the virtual patient or an event comes from the simulation tool that mentions that the virtual patient arrived first.

For simplicity and to avoid repetition, different scenarios can be seen in this GRAFNet, such as the following:

- If the real patient is faster than the virtual patient. After step 9, the token will follow one of these two paths (steps):
  - 3, 15, 6 and return to step 9
  - 4, 7 and sink (remove the token from the GRAFNet, which means the corresponding real patient has left the hospital)
- If the real patient is slower than the virtual patient. After step 9, the token will follow one of these three paths (steps):
  - 19, 27, 20, 26, 6 and return to step 9 (the real patient and the virtual patient arrive at different activities)
  - 19, 27, 26, 6 and return to step 9 (the real patient and the virtual patient arrive at the same activity)
  - 19, 2, 5 and sink (removes the token from the GRAFNet, which means the corresponding real patient has left the hospital)

After describing the concepts behind the execution of the real world and the virtual world GRAFNet patient pathways, the synchronization between the two GRAFNets will be summarized in the following example:

- In the real GRAFNet patient pathways: when the transition RP_arrive_Entrance detects a patient arriving at the hospital entrance, the condition associated with this transition will be changed to true. A token will be created at step 13, indicating that a real patient has arrived at hospital entrance. Thus, the state of step 13 will change to active. Due to the activation of step 13, the action associated with this step will be executed to inform the virtual patient pathways that a new patient has arrived at the hospital entrance, as illustrated in Figure 4.21, whereas the token that is in step 13 will move to step 14 to activate it. The action associated with this step will be executed to illustrate the moves of the patients in the real patient pathway.

  Notice: the state of the real patient is moving, as depicted in Figure 4.22.

- In the virtual GRAFNet patient pathways: because the virtual patient pathways have received an event from the real patient pathways, this indicates that a real patient has arrived at the entrance of the hospital, and the condition associated with the transition Waiting_RP_Entrance_Event will change to true, as illustrated in Figure 4.21. As a result, the token will be created at step 10. Thus, the state of step 10 will change to active. Due to the activation of step 10, the action associated with this step will be executed to create a virtual patient at the entrance of the virtual hospital. In this case, the same token in step 10 will be representing this created patient. This process will take a very short period of time. After that, the token will move from step 10 to step 6 to assign a random path for the created patient to follow. In this case, step 10 will be deactivated and step 6 will be activated for a short delay. Then the created patient will be released to follow the proposed path when the token moves from step 6 to step 9. As a result, step 6 will be deactivated and step 9 will be activated until an event comes from the real patient pathway that mentions a real patient arriving at the activity or exit door faster than the virtual patient (right branch) or an event comes from the simulation tool that mentions that the virtual patient has arrived first (left branch).

  Notice: the state of the virtual patient is moving, as depicted in Figure 4.22.

- In the virtual GRAFNet patient pathways: Suppose the virtual patient arrives first at the activity. This means the condition associated with the transition Waiting_VP_Arrive_Event will change to true. Thus, the token passes from step 9 to step 19. This leads to the deactivation of step 9 and the activation of step 19. As a result, the action associated with step 19 will be performed to demonstrate the process of preventing/blocking the virtual patient from beginning the activity until an event from the real patient pathway indicates that a real patient has arrived at the activity (left branch) or the exit door (right branch), as demonstrated in Figure 4.23.

  Notice: the state of the virtual patient is blocked, whereas the real patient is moving, as depicted in Figure 4.22.

- In the real GRAFNet patient pathways: Assume that the real patient arrives at the activity. The condition associated with the transition RP_Arrive_Event is changed to true, and the token moves from step 14 to step 17. As a result, step 14 is deactivated and step 17 is activated. The action associated with step 17 will be carried out to notify the DT that the patient has arrived at the activity, as shown in Figure 4.24.

- In the virtual GRAFNet patient pathways: The condition associated with the transition Waiting_RP_Arrive_Event will change to true, as depicted in Figure 4.24.

- In the real GRAFNet patient pathways: The receptivity following step 17 is equivalent to 1. This means the condition is always true. Thus, the token will move to step 18 immediately. This leads to the deactivation of step 17 and the activation of step 18. The action that is associated with step 18 will be executed to illustrate that the patient is inside the activity, as illustrated in Figure 4.25.

- In the virtual GRAFNet patient pathways: from the point of view of the virtual patient pathways, the real patient has two options: either arrive at the same activity where the virtual patient is blocked (right branch), or arrive at a different activity (left branch), as shown in Figure 4.25.

  Notice: the real patient exists inside the activity, while the virtual patient pathway tries to determine the location of this patient.

- In the virtual GRAFNet patient pathways: Assume that the virtual patient pathways determines that the real patient has arrived at the different activity, due to an event received from the simulation tool. In this case, the condition associated with the transition Arrive_Diff_Activity will change to true. As a result, the token will move from step 27 to step 20. This leads to the deactivation of step 27 and the activation of step 20. The action associated with step 20 will be executed to accelerate the virtual patient to the same location as the real patient. Then, the token will move from step 20 to step 26. This leads to the deactivation of step 20 and the activation of step 26. After that, the action associated with step 26 will be executed to illustrate that the virtual patient is performing the same activity as the real patient, as demonstrated in Figure 4.26. The virtual patient will continue performing the activity until a new event comes from the real patient pathways that mention the real patient finishing the activity.

  Notice: the real patient and the virtual patient are performing the same activity.

- In the real GRAFNet patient pathways: Assume the transition RP_leave_Event detects a patient finishing the activity. The condition associated with this transition is changed to true. As a result, the token will move from step 18 to step 22. This leads to the deactivation of step 18 and the activation of step 22. The action associated with step 22 will be executed to inform the virtual patient pathways that the real patient has finished the activity, as demonstrated in Figure 4.27. After that, the token will return to step 14. This leads to the deactivation of step 22 and the activation of step

14, as in the previous Figure 4.22. The action associated with step 14 will be executed to illustrate the moving process of the patient. After that, the patient can decide to go to the next activity or to leave the hospital.

- In the virtual GRAFNet patient pathways: Because the virtual patient pathways receive an event that indicates that the real patient has finished the activity, the condition associated with the transition Waiting_RP_leave_Event will change to true. As a result, the token will move from step 26 to step 6 to repeat the same process: assign a path for the virtual patient to follow, and then release this patient to move, until an event comes from the real patient pathways that mentions a real patient arriving at the activity or the exit door faster than the virtual patient or an event comes from the simulation tool that mentions that the virtual patient has arrived first, as shown in the previous Figure 4.22.

  Notice: The real patient and the virtual patient return to the moving activity.

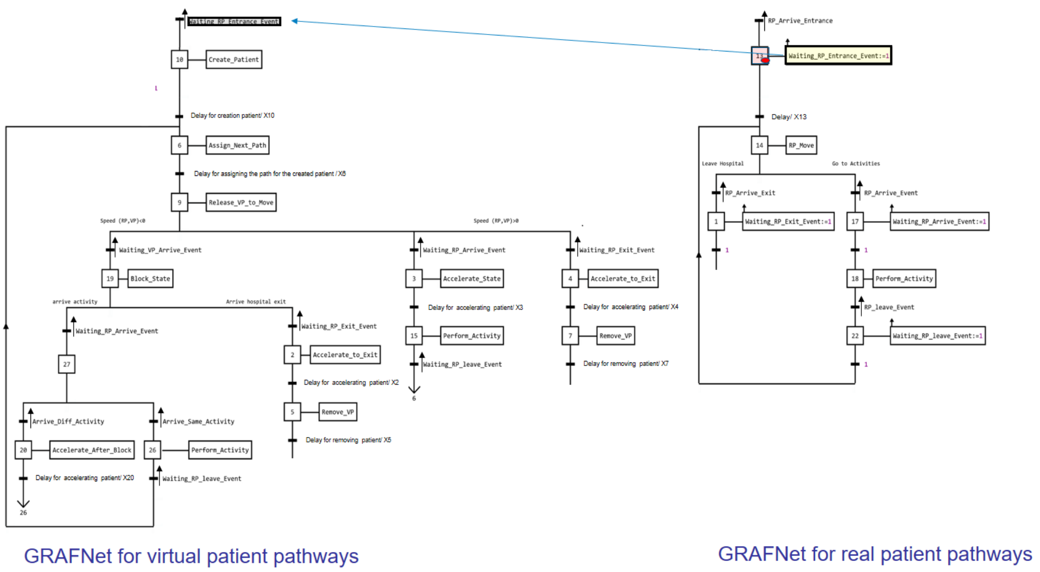


Figure 4.21: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 1, see Figure 6.8

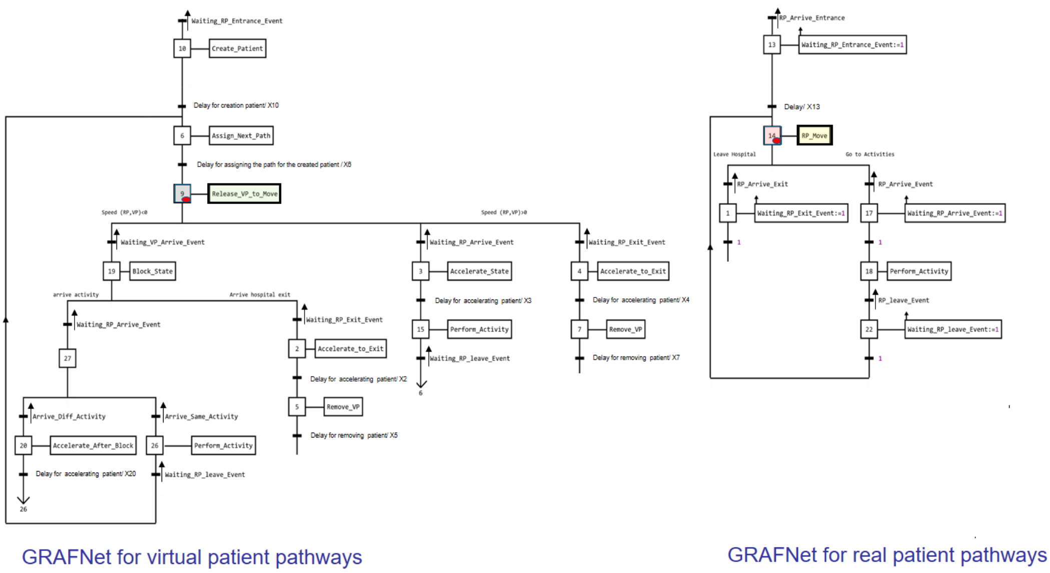


Figure 4.22: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 2, see Figure 6.9

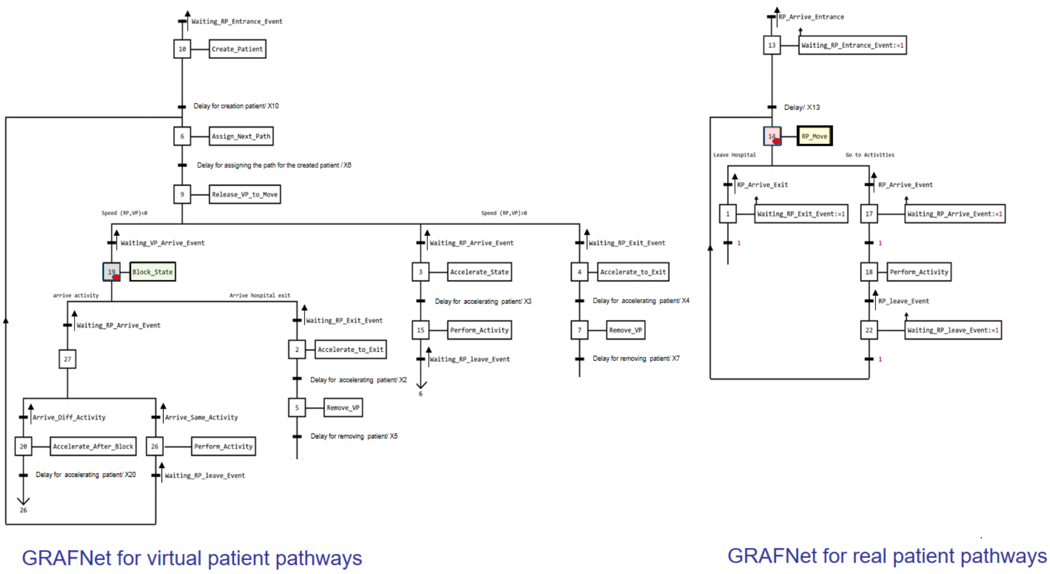


Figure 4.23: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 3, see Figure 6.10

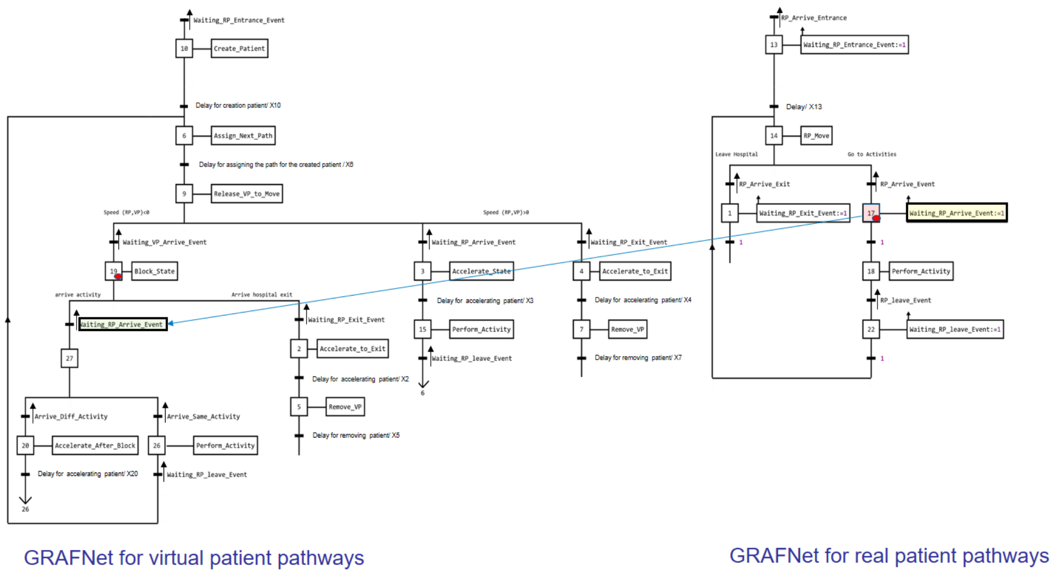


Figure 4.24: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 4, see Figure 6.11

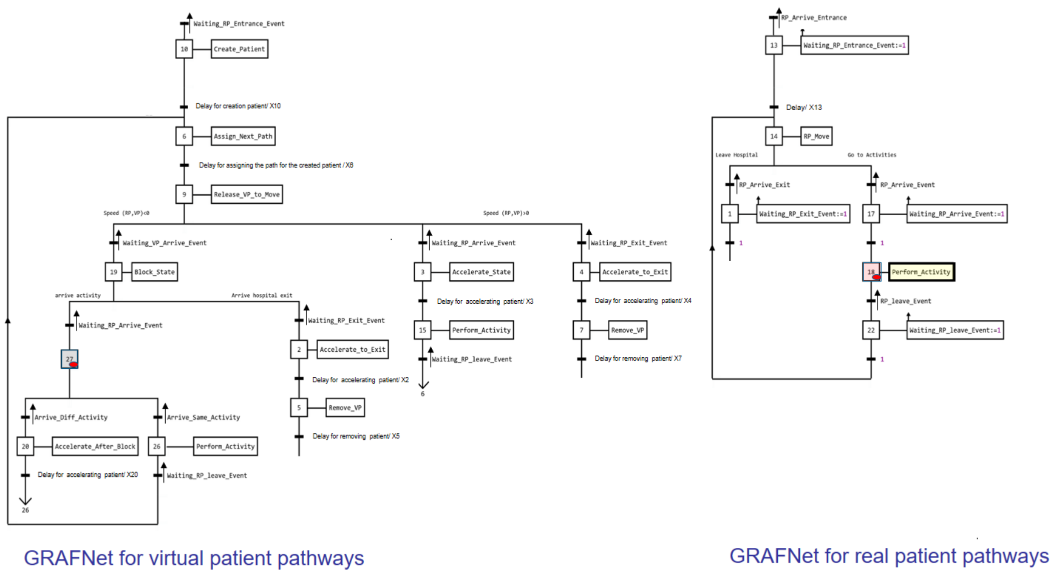


Figure 4.25: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 5, see Figure 6.12

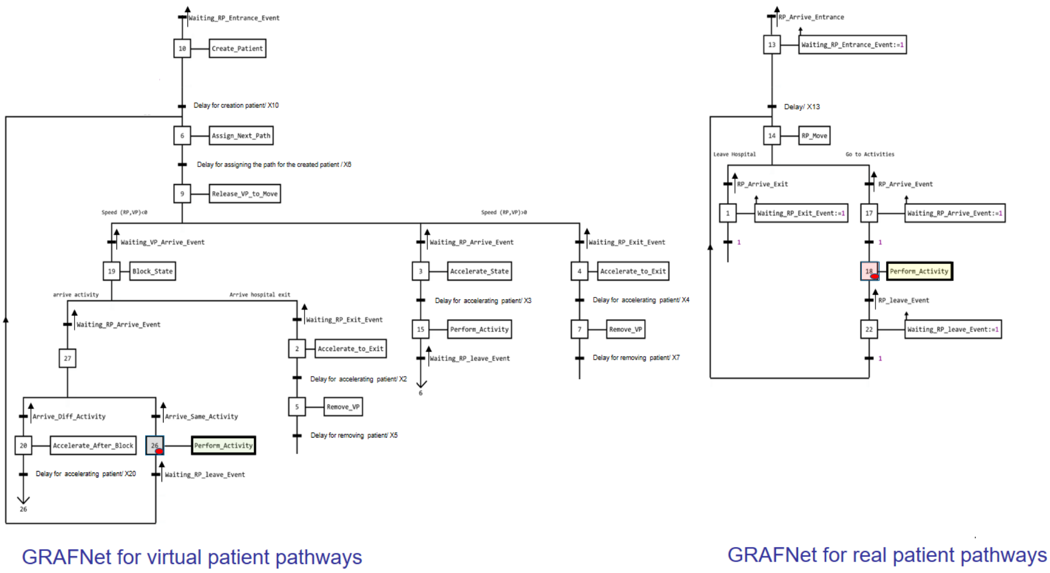


Figure 4.26: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 6, see Figure 6.13

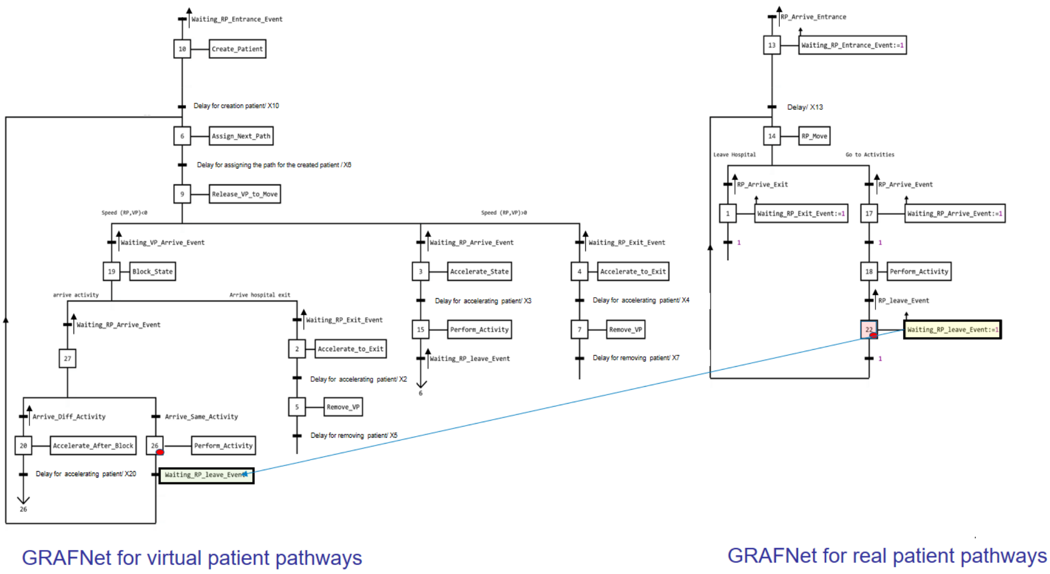


Figure 4.27: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 7, see Figure 6.14

In short, from the examples above, we can see the following:

1. The structure of the GRAFNet real patient pathways is not exactly identical to the structure of the GRAFNet virtual patient pathways (e.g., different numbers of steps, different types of actions, etc.).
2. One step in the real patient pathways can correspond to multiple steps in the virtual patient pathways during synchronization. For example, when the sensor detects the patient at the entrance door in the real patient pathways, the response from the virtual patient pathways will be as follows: (1) create patient, (2) assign path to the created patient, and (3) release the created patient to move.
3. In the virtual patient pathways during synchronization, there is sometimes a need to block or accelerate the virtual patient, whereas these actions do not exist in the real patient pathways.
4. Different types of synchronization events have to be considered in the GRAFNet virtual patient pathways. Is this event, for example, from the real patient pathways (external event) or from the simulation tool event (internal event)?

4.4.4 Virtual control: an event handler to allow the synchronization between the DT of patient pathways and real patient pathways based on DES

Virtual control is an event handler component that must be included in the DTM to make the synchronization between this DT and the real world possible. For each sensor in the real world, an event handler component corresponding to it must be implemented in DTM. The main purpose of this component is to receive the real-world events that is captured by the sensors and based on these events, actions can be executed in the DTM. For example, when the event handler receives an event that a real patient arrives at the entrance of the hospital, this event handler must create a virtual patient that represents the real patient that arrives at the real hospital. Moreover, when the event handler receives an event that the real patient arrives at activity while the virtual patient is still moving, the event handler will accelerate the virtual patient to the same location as the real patient. Furthermore, when the event handler receives action that mentions the virtual patient arriving at the activity while the real patient is not, the event handler will block the virtual patient corresponding to this real patient from starting the activity until the real patient arrives at the same activity. Last but not least, when the event handler receives an event that the real patient leaves the hospital, the event handler will remove the corresponding virtual patient from the DTM hospital. A question such as: “Based on what the event handler can know, what type of action must be executed (create, block, accelerate, remove the virtual patient) and who the patient must be involved in this action?” can be raised here. To answer this question, some attributes have been assigned to each virtual patient among them: Patient ID, Patient current location. Patient ID will be used to be a reference for the real patient, as will the patient’s current location will be used to store the location of the real patient. The event handler component can access the ID of the real Patient to know who is the virtual patient must be involved in the executed action (e.g., accelerate virtual patient whose ID is 1234), as well the event handler will use the location of the real patient to know where to move the virtual patient. Other attributes will be involved in this process for example born attribute, custom speed, teleportation speed, and patient next location. These attributes will be discussed in detail in section 5.5.1.

To implement the event handlers, we propose using a publish/subscribe mechanism. Each virtual control will subscribe to a broker channel, where the sensor can publish the events as illustrated in Section 3.4.3. When the event handler receives the event, the event handler will extract the real patient ID, as well as the current location for this patient (architectural item where the sensor is attached) from the event. Based on these two types of information, the event handler will locate the virtual patient that has the same ID as the real patient. Based on the location of the real patient, the event handler can execute an action. For example, if the event handler receives an event from the sensor that a real patient at the entrance, for sure the action must be to create a virtual patient because this real patient has no virtual patient refer to him/her in the DTM, whereas if the event handler receives an event that indicates that the real patient arrives at for example activity X. That means the virtual patient exists in the DTM because this is not an entrance event. Based on this, the event handler will locate the virtual patient corresponding to the real patient using the real patient ID, and then it will move him/her to the same location as the real patient and so on. However, all of these actions are illustrated in detail using the GRAFNet chart in Section 4.4.3, and the implementation of these actions has been details in Algorithms. 2 and 3

4.5 Comparison of the Proposed Approach with the Reviewed Approaches

This section illustrates and details the comparison of our proposed approach with the other initializing and synchronizing approaches that were reviewed in Section 2.4.8. Table 4.3 summarizes the results of this comparison in terms of the following attributes:

- Real-time synchronization: At each event detected in the real world, the DTM's behavior must be close enough to reality. For example, the number of patients in the virtual hospital and the real hospital are equal at each activity; if a real patient moves, a virtual twin corresponding to the real patient moves as well, even if they are moving in different directions, and so on.
- Real-time prediction: The ability to initialize and run the DTP with the most recently updated data (most recently updated dynamic distribution) in order to anticipate the future. In fact, the prediction in a traditional simulation model is started with an empty state, and the input for these simulations is historical data and the distributions used are static, which is not the case in our work. Our predicting model starts with the current state of the patient pathways and the distributions used are updated empirical distributions.
- Maintain availability in case of problem: The ability of a DT to keep running even if one of its servers fails. Similarly, the ability to restart the DT in the same state as the real world in case the DT is turned off for whatever reason.
- Backup data: The ability to create a copy of real world events so they can be easily recovered if the original data is lost or corrupted.
- Possibility of running the replay model: The ability to run the DT based on past events. This feature can be used to analyze hospital performance, find bottlenecks in patient pathways, discover pathway complexities and service dependencies, discover different patient pathways, and detect deviations between the real world and the DT, among other things.
- Maintain a DTM: This feature allows the current location of each patient at the hospital to be tracked, including the location of patients between two sensors that the database does not show. Similarly, it may be possible to detect the current deviation between the real and virtual worlds and determine whether the real or virtual patient is on the wrong path.

- Reduce overall complexity of the simulation system: Maintaining a DTM to continuously monitor the real world can overload the system and consume system resources. For this work, we are attempting to run the DTM on demand rather than continuously.

Attribute / Approach	HospiT'Win	State Collection Approach	Base Simulation Approach
Real-time synchronization	Supports	Does not support (periodic approach)	Supports
Real-time prediction	Supports	Does not support (does not update during prediction)	Does not support (does not update during prediction)
Maintain availability in case of problem	Supports	Supports	Does not support
Backup Data	Supports	Supports	Does not support
Possibility of running replay model	Supports	Does not support (does not store all of the events)	Does not support (there is no database)
Maintain emulator for monitoring	On Demand	Does not support (No need)	Mandatory
Reduce overall complexity of the simulation system	Supports	Supports	Does not support

Table 4.3: HospiT'Win compared with other approaches

4.6 Conclusion

This chapter has discussed the various concepts behind the initialization of the DT models of patient pathways with the current state of the real patient pathways. It has also highlighted the different parameters to be taken into account when initializing the DTM and the DTP; for example, the number of patients and their locations (in which activity). These two parameters can be appropriate for the DTM, whereas a third parameter will be included when initializing the DTP, which is the amount of time the patient spends at each activity. Furthermore, this chapter mentions that the DTM will be used to initialize the DTP, whereas the DTM will be initialized by the database, and will stay synchronized with the real world.

Different issues related to synchronization have been discussed in detail. For example: the real patient is faster than the virtual patient, the virtual patient is faster than the real patient, the real patient and the virtual patient move in two different directions. Due to the fact that the virtual patient does not know the next destination to follow after his/her creation or at the time he/she completes a specific activity, a random path is selected using historical data. This path will be assigned to the patient and at this stage, we assume the assigned path is correct, except in the case where the two patients (the real one and virtual one) take two different directions. In this case, the synchronization phase will adjust the selected virtual path so that the virtual patient will follow the real patient. For synchronizing the simulation clock with the real-world wall clock, this chapter mentions that the clock is considered an important part of the synchronization for the DTM, whereas the clock is considered an important part of the initialization for the DTP. Initializing the simulation clock in the DTP means that "from this current time, I would like to know the state of the hospital after X units of time". Synchronizing the clock in DTM means that "at this moment, I would like to see the current state of the hospital".

To illustrate all of the concepts and the actions behind our proposed approach for the synchronization, the GRAFNet chart has been developed by merging some of the logic from the Petri Net and the GRAFCET models, due to the limitations of these two modeling languages. This chart has been used to show the different concepts behind synchronizing the DTM with real patient pathways. First, a generic GRAFNet for the real patient pathways, as well as a generic GRAFNet for the virtual patient pathways, has been discussed in detail. After that, the synchronization process between the two GRAFNets was illustrated.

Finally, a comparison of our initialization and synchronization approach and the other proposed approaches was presented at the end of this chapter. Based on this comparison, it seems that only our approach covers different real-time synchronization issues, compared with the other approaches.

5

Proof of Concept: Monitoring and Predicting Models

5.1 Introduction

A proposed methodology for designing a Digital Twin (DT) for patient pathways was detailed in Chapter 3. Initializing the DTs (Digital Twin for Monitoring (DTM) and Digital Twin for Predicting (DTP)) to have the same state as the real world and synchronizing the (DTM) with the real world at each detected event was discussed in Chapter 4. To validate our methodology and to verify the initializing and synchronizing algorithms, an experimental platform for a basic use case (an outpatient department in a hospital) has been developed. This platform is a composite of two primary components: an emulator and DTs (DTM, DTP). Indeed, due to different constraints such as social issues, technical issues, privacy issues, ethical issues and research experiment issues, the real hospital was replaced by an emulator for generating the same events as those that are in the real hospital. Moreover, nowadays in France it is difficult to find a real hospital that deploys sensors to collect real-time data. More details about this platform will be discussed in this chapter. Regarding the use case, it includes five activities: a Waiting Line (WL), a Registration Desk (RD), a Waiting Room (WR), and two Exam Rooms (ER1, ER2). The aim of this chapter is to follow the methodology proposed in Chapter 3 and use the initialization and synchronization algorithms that were discussed in Chapter 4 to design a proof of concept for the DTM and a prototype for the DTP. To achieve this goal, the experimental platform has been used, but in real life it would be easy to switch the emulator to the real hospital for making a connection to the DT, as will be explained in this chapter. To achieve the aims of this chapter, different requirements are needed:

- The developed emulator must conform to a meta-model of real patient pathways, discussed in Section 3.4.1.1. In other words, the emulator should include the same elements proposed in the real patient pathway meta-model in Section 3.4.1.1. In this research, the emulator will replace a real world hospital.
- The emulator must be as close as possible to the real hospital. For example, it must include different patient pathways, different speeds of patients, and different durations of activities. Furthermore, the emulator must mimic the different behaviors of patients, such as those who are dissatisfied and leave the hospital without seeing a doctor, those who are satisfied and wait for the doctor, those who come merely for an appointment, those who come for treatment, and so on.
- A meta-model for developing the DT of patient pathways is required. In fact, the DT that we are going to develop needs a variety of elements, including real-world elements, Discrete Event Simulation (DES) elements, and Internet of Things (IoT) features. In order to connect the virtual and the real worlds, all of these elements and the relation between them are highlighted in the meta-model that was developed in Section 3.4.1.2.
- A knowledge model that captures as much information as possible about the real hospital is needed. Indeed, this knowledge model will be used to develop the offline and the online simulation models. To meet this demand, we have designed rich process flow modeling elements, which are mentioned in Section 3.4.1.3

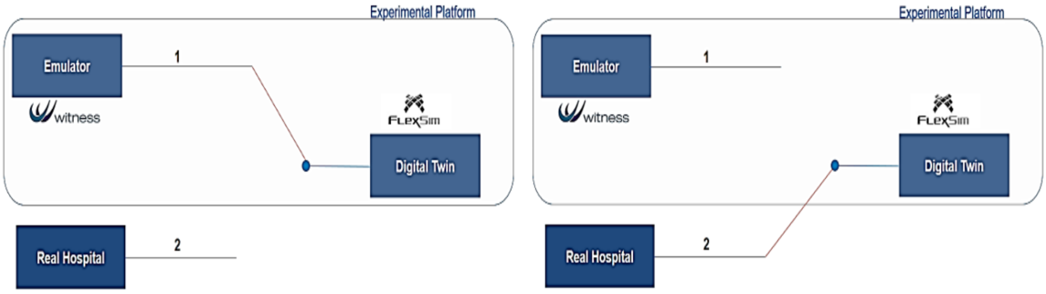
5.2 Experimental Platform

In this research, the experimental platform is considered a testbed for testing the proposed DT models (DTM and DTP). This platform consists of two components: (1) an emulator which was designed and developed with the WITNESS simulation tool[1], and (2) a DT which was designed and developed with the FlexSim simulation tool [2], as illustrated in Figure 5.1. The emulator is built to emulate a fictional hospital due to the difficulty of finding a real hospital that deploys sensors to collect real-time data for feeding the DT. Even if this type of connected hospital becomes real in the near future, we believe that the experimental platform remains a relevant component because it can be used to test and validate the structure, the behavior and the control feedback of the developed DT, as well as the cybersecurity issues regarding connection of the DT with the real world. The testing and the validation process with an emulator will take place without interrupting, affecting or disrupting the daily activities of the hospital. Following the testing and the validation process, it will be simple to switch the connection by disconnecting the DT from the emulator and connecting it to the real hospital, as illustrated in Figure 5.1 (2).

Figure 5.1 (1) shows the main components of the experimental platform, where the DT connects with the emulator, and Figure 5.1 (2) shows the connection between the DT and the real world.

Regarding the development of the emulator, an architectural layout of the use case has been used inside it. This layout describes the building structure of this hospital. It also describes a set of five activities {WL, RD, WR, ER1, ER2} and their locations in the hospital, as shown in Figure 5.2 (a). To reflect reality, the emulator has used stochastic distributions (1) to generate patient arrivals in different time slots, (2) to set the duration of each activity, and (3) to move patients at different speeds (fast, slow, and average) through different pathways, as depicted in Figure 5.2 (b). Moreover, this emulator was developed using a DES tool because the patient will stop at different discrete points where different activities exist (e.g., the patient might stop at the registration desk to complete the registration process, or the patient

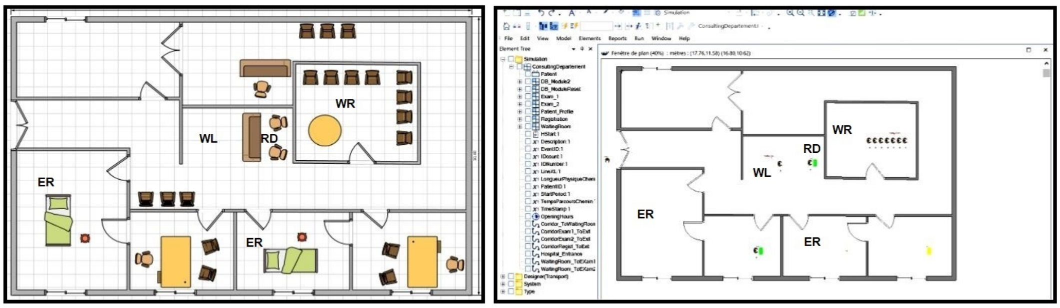
[1] Commercial software from the Lanner company, https://www.lanner.com
[2] Commercial software from the FlexSim company, https://www.flexsim.com



(1) Digital twin connected with an emulator **(2) Digital twin connected with the real hospital**

Figure 5.1: Experimental platform main components

might stop in the waiting room to wait, and so on). For us, even though the emulator was developed with a discrete tool, its behavior is similar to the real world (continuous behaviour). For example, different patients during different times enter the hospital following different pathways to have treatment. The clock of the emulator is adjusted to move continuously, like the real-world wall clock, etc.



(a) Architectural layout with fixed resources (b) Emulator

WL : Waiting Line
ER : Exam Room
WL: Waiting Line
RD : Registration Desk

Figure 5.2: Emulator with an architectural layout

In the emulator, there are various types of patients moving from the source to the sink. The source is used to generate patients inside the model (called hospital entrance), and the sink is used to remove patients from the model (called hospital exit). Conforming to the meta-model in Figure 3.2, during the movement of the patients from the hospital entrance to the hospital exit, there is a sequence of activities that the patient may visit. Each activity takes place in an architectural item, such as a room or in an open area (e.g., WL and RD exist in an open area, whereas WR, and ER exist in a closed area called room). Each architectural item is attached to a virtual sensor in the emulator to detect the patient when he or she arrives at the activity, and when he or she leaves the activity. At each sensor, there is a database query that writes the event information to the database. This information represents the ID of the patient detected by this sensor, the ID of the sensor that detected this patient (called EventID), and the detection time (time stamp for this event).

To reflect reality, the emulator has been run to generate data for two months. These data will be used as historical data in our experiment. The remaining sections of this chapter depict the entire process of creating the DTM and DTP, beginning with the design phase and ending with the running phase, as illustrated in Figure 5.3.

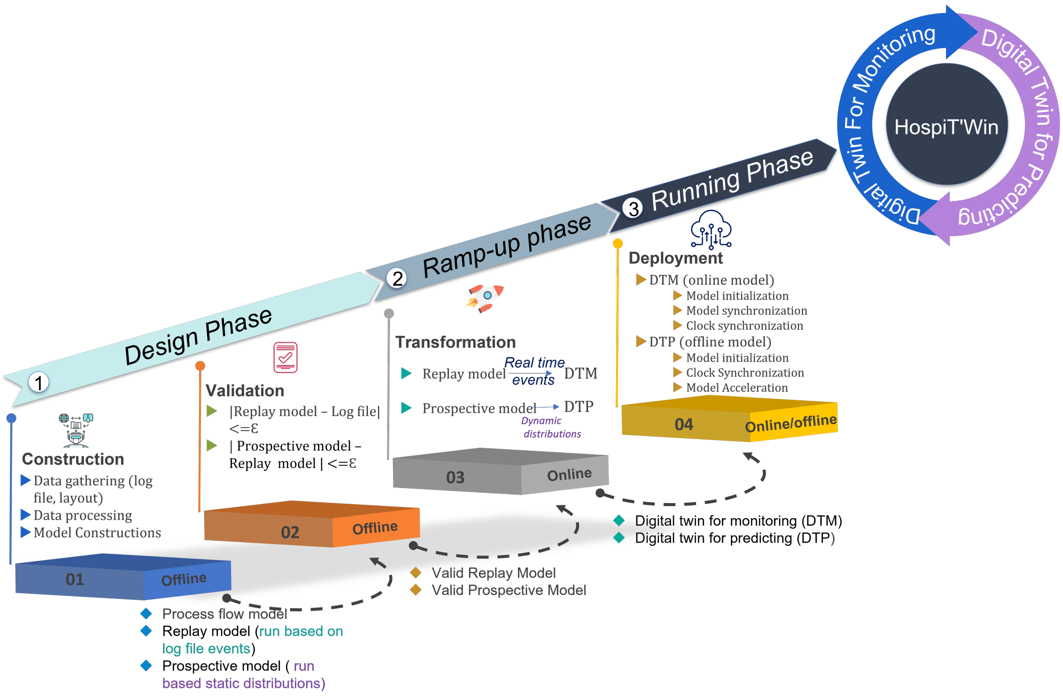


Figure 5.3: Proposed methodology for designing and building HospiT'Win

5.3 Design Phase

According to the proposed methodology in Figure 5.3, the first phase is the design phase. In this phase, there are two steps : construction step and validation step as will be discussed in sections 5.3.1 and 5.3.2.

5.3.1 Construction

The first step in the construction is gathering historical data to construct a process flow model, a replay model, and a prospective model as will be discussed in Sections 5.3.1.1, 5.3.1.2, and 5.3.1.3.

5.3.1.1 Developing a process flow model

The developed emulator has been run to generate a log file of events for a significant period of two months, following this format {PatientID, TimeStamp, Event}. PatientID represents the identification of the patient (e.g., 2019.1.2.8.6.2), TimeStamp identifies when a specific event occurs (e.g., 02/01/2019 8:06:02 AM), and Event is a short description of the event (e.g., Patient arriving: ConsultingDepartment.Hospital_Entrance). Figure 5.4 illustrates a sample of this log file. These events have been used to construct a process flow model, a replay model, and a prospective model.

First of all, this log file has been cleaned of anomalies and outliers. For example, there are some patients who spent 20 minutes at the registration, compared to the maximum duration time of registration, which is around 7 minutes. This is an outlier, as there were only 3 cases out of more than 6,000 patients. In another case, we found 2 patients that were with the doctor in the exam room for more than 5 hours, compared to the discovered maximum time for the exam room which was around 30 minutes, etc. These outliers were deleted. Then, as stated in Section 3.4.1, this log file must be structured in a format that

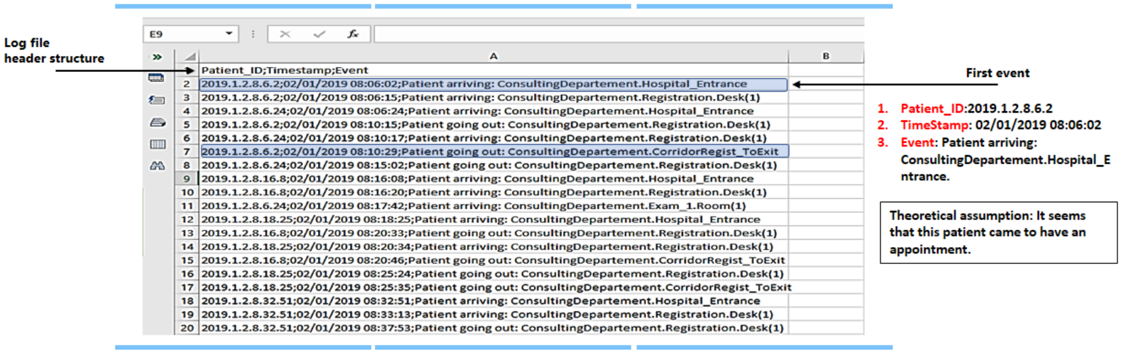


Figure 5.4: A sample of the unstructured log file that was created

is compatible with a Process Mining tool. Indeed, a Process Mining tool was used (in this work we used Disco (Fluxicon, 2021)) to discover the different patient pathways, such as {PatientID, ActivityName, StartTime, EndTime}. A sample for the structured log file is shown in Figure 5.5.

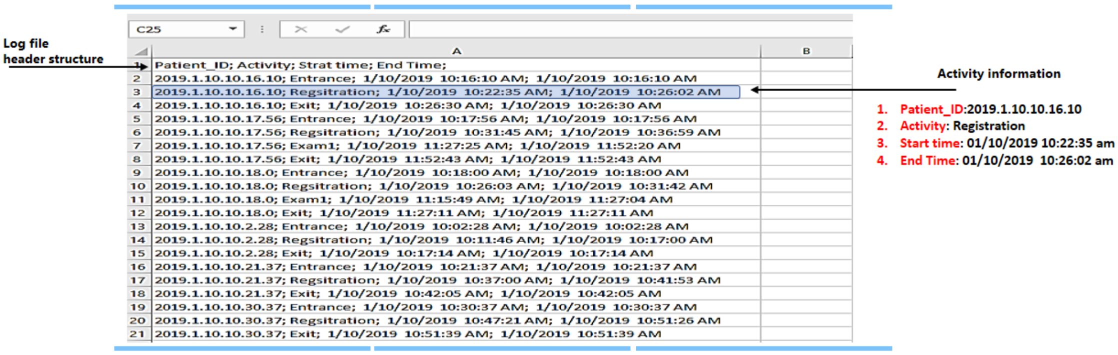


Figure 5.5: A sample of the structured log file that was created

In Figure 5.6, the process mining tool has used the structured log file as an input to discover the various patient pathways the patient may follow in the hospital. These pathways will be used to design the process flow model that is illustrated in Figure 5.7.

It is important to mention here that complete data availability is required. In other words, if the data is complete and can describe the different pathways, this will provide a more accurate process flow model. As a result, a more accurate simulation model will be developed; but if the data is not complete, some pathways will not be discovered with the process mining tool. This will lead to an inaccurate process flow. In this case, more analyses and hypotheses must be made to solve this problem. In this work, we assume that the data is complete and the process flow model is valid and includes all of the different pathways that a patient may follow.

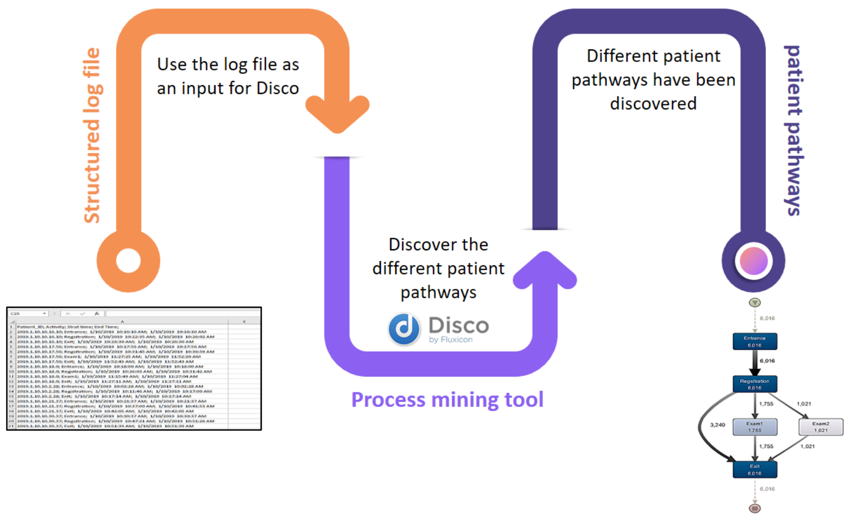


Figure 5.6: Discovering the patient pathways from the log file

Figure 5.7 shows a process flow model which is a knowledge model that illustrates the different pathways, including the different activities the patients may visit, during their journey to have treatment. This model was designed using the process flow elements that have been discussed in Section 3.4.1.3. Different analyses for the structured log file have been performed to find the different durations for each activity.

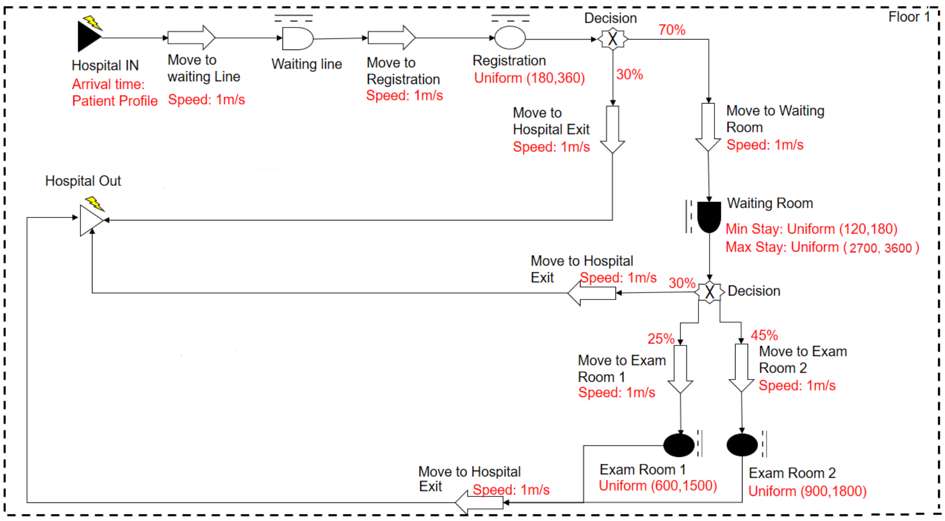


Figure 5.7: Process flow model

For simplicity and to make the model more readable, Table 5.1 illustrates different data that describe the process flow model depicted in Figure 5.7. This table illustrates the different pathways the patient may follow from entrance to exit, as follows: Path (1) corresponds to a patient who comes to the hospital for appointments. Path (2) corresponds to a patient who waits too long in the waiting room and decides to go back home. Path (3) and path (4) correspond to a patient who has medical treatment and then goes home. Also, the average distances that the patient may walk during his/her treatment are depicted in the same table. Furthermore, based on theoretical computation, the average walking speed of patients is 1

Path ID	Sequence of activities	Average distance (meter)	Average duration time (second)
Path (1)	Entrance → Waiting Line → Registration → Exit	20	20
Path (2)	Entrance → Waiting Line → Registration → Waiting Room →Exit	32	32
Path (3)	Entrance → Waiting Line → Registration → Waiting Room →Exam Room 1 → Exit	29.6	29.6
Path (4)	Entrance → Waiting Line → Registration → Waiting Room →Exam Room 2 → Exit	30.5	30.5

Table 5.1: Information related to the process model

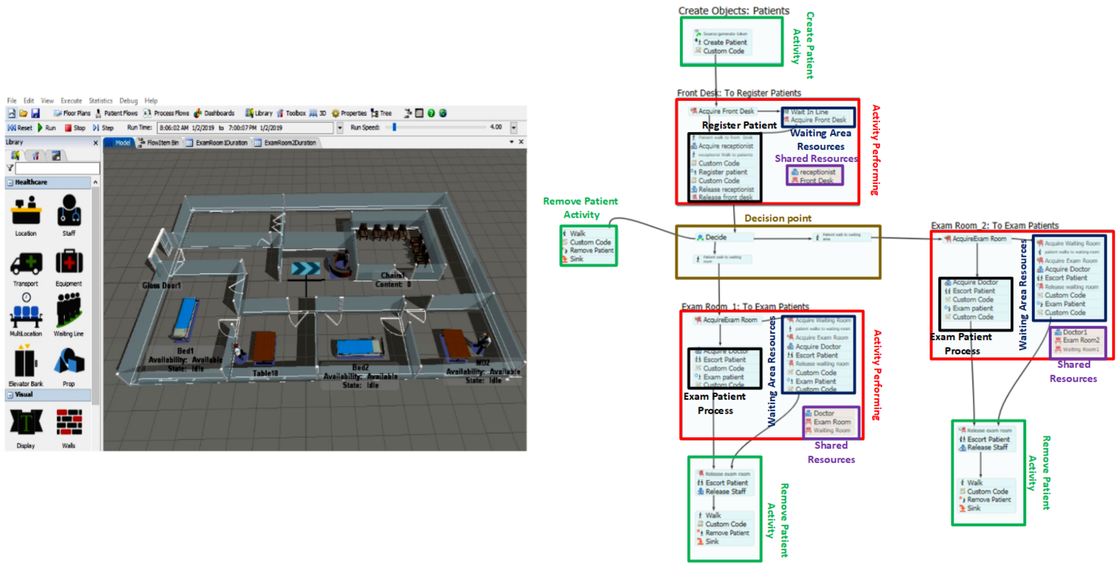
m/s. Thus, the duration for each path, starting with entrance and ending with discharge, is calculated and presented in the same table.

After developing the process flow model, the second step is to develop the offline simulation models (the replay model and the prospective model). The output knowledge from the process flow model will be used as input data for creating these simulation models, as illustrated in Figure 3.5, step 3.

5.3.1.2 Developing a replay model

According to the construction step, Section 3.4.1.4 in our proposed methodology, the replay model is the first model that will be developed after the process flow model. In other words, the replay model must include all the activities, pathways, patient speeds, etc., which are contained in the process flow model. For the statistics part and the synchronization part that are shown in the process flow model, they will be used for the prospective model and the DTM model respectively, not for the replay model. In fact, the replay model has been used to study patient behavior in the past and to check model structure. However, this model is somewhat similar to the model that was discovered using Disco, except that this model includes additional knowledge to run on top of the DES tool (e.g., 3D objects). For example, patients come to the hospital corresponding to the arrival times that exist in the log file, and the duration time for each activity for each patient must be the same duration time as in the log file. In addition, the virtual patient must follow the same pathways that were discovered in the log file. Furthermore, the building of the replay model must conform with the DT meta-model that was discussed in Section 3.3. This means that the replay model elements and the relations between these elements must follow DT meta-model rules.

Figure 5.8 (a) depicts the replay model with 3D-geometric objects, whereas Figure 5.8 (b) illustrates the FlexSim process flow for this model. As shown in these figures, there are different types of activities which the patient can visit. First, the patient will be created at the entrance door as illustrated in this model; the creation time for the patient will be the same arrival time as registered in the log file. Then the patient will move to the registration desk through a waiting line. The duration time for registration is the same as the registered duration in the log file. After that, there is a gateway (decision point), where the patient can go home or go to the waiting room. Then, when the patient has finished waiting, the patient may go to the exam room 1, exam room 2, or go home, as there is a gateway there. Finally, the patient who is in the exam room can go home once the treatment is finished.



(a) Replay Simulation Model (b) FlexSim Process Flow Model

Figure 5.8: Replay model

5.3.1.3 Developing a prospective model

The second offline simulation model that will be created after the replay model is a prospective model. This model must conform with the developed process flow in Figure 5.7 in addition to the statistics part except the synchronization part, it will be used with DTM. The statistics part that appears in the process flow model has been discovered from the log file generated by the emulator. In fact, to get these statistics, we have analyzed the structured log file. We found that the duration time for the registration activity follows a uniform distribution, represented by **Uniform (180,360)**, and that waiting room has two static distributions. The first represents the maximum duration time that the patient waits before deciding to go home, **Uniform (2700,3600)**, for whatever reason. The second distribution, which represents the minimum time the patient waits before seeing a doctor, is represented by **Uniform (120,180)**. However, all of the discovered durations are depicted in the process flow Figure 5.7, and all of the aforementioned durations are in seconds.

For the 3D design and the process flow, the replay model and the prospective model are the same. One of the main differences between these two models is that the replay model is controlled by a specific period of time. In other words, the start point of the replay model is the start date in the log file, and the model continues running until reaching the last date in the log file (which contains data for a two-month period). The prospective model is not limited to any date and time. It could be run from the past and accelerated to the near future, as explained in Section 3.4.1.4.

5.3.2 Validation: replay and prospective models

The second step of the design phase is the validation of the replay and prospective models. To perform the validation, we chose the indicators shown in Table 5.2 and used the validation algorithm 1 shown in Section 3.4.1.5.

Indicators	Description
ind_1	Number of incoming patients at hospital
ind_2	Number of outgoing patients from hospital
ind_3	The number of incoming patients at activity i
ind_4	Number of outgoing patients from activity i
ind_5	Minimum duration time for activity i
ind_6	Maximum duration time for activity i
ind_7	Number of patients in the process of activity i
ind_8	Average duration time at activity i
ind_9	Average number of patients at activity i
ind_10	Average length of stay

Table 5.2: Indicators for validating the models

Based on a selected set of common indicators[3], the comparison between the events generated from the replay model with the events stored in the log file has been made, as illustrated in Figure 5.9 (a). To validate the prospective simulation model, the replay model has been used as a reference model, and the comparison between the events generated from the replay model and the events generated from the prospective model has been made based on a selected set of indicators, as illustrated in Figure 5.9 (b).

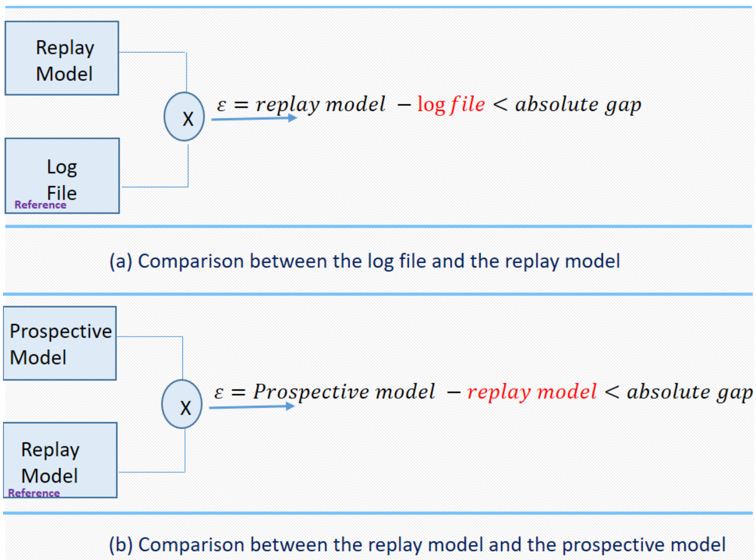


Figure 5.9: Model validation indicators

In this work, the model is considered valid if the difference between the reference model and the model to validate has a threshold value within this range [-5%, +5%][4]. For example, if

[3]These indicators have been selected to be used in our comparison algorithm. These indicators could be expanded or shrunk, depending on the decision-maker.

[4]We used the threshold value as an example in order to make the comparison. Although this value may not reflect reality, we have found it to be appropriate to our work. This value could be different in

the incoming number of patients in the replay model over the two months is 962, and the incoming number of patients in the prospective model for the same period is 942, we consider that for this indicator, the prospective model is valid because the difference between the number of the incoming patients in the two models is within [914, 1010], which is acceptable compared with the predefined threshold range. In this work, we have checked all of the predefined indicators to test the validity of the replay and the prospective models. As a result, our models pass the validation test. Tables 5.3 and 5.4 illustrate a summary of our results for three replications. In other words, the replay model has been run three times, and the statistical average of these three runs has been compared with the statistics that came from the log file. The same process has been applied to the prospective model.

1/2/2019 8:06:02 AM to 1/2/2019 6:27:08 PM
The time in seconds

(Hospital)	Log File	Replay Model	Ratio	Max Range	Min Range	Accepted Error	Check
Number of incoming patients	104	104	1	109.2	98.8	0.05	Yes
Number of out going patients	104	104	1	109.2	98.8	0.05	Yes

(Exam Room_1)	Log File	Replay Model	Ratio	Max Range	Min Range	Accepted Error	Check
Incoming Nb of patients	27	27	1	28.35	25.65	0.05	Yes
Outgoing Nb of Patients	27	27	1	28.35	25.65	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Min Time	621	621	1	652.05	589.95	0.05	Yes
Max Time	1479	1479	1	1552.95	1405.05	0.05	Yes

(Exam Room_2)	Log file	Replay model	Ratio	Max range	Min range	Accepted error	Check
Incoming Nb of patients	16	16	1	16.8	15.2	0.05	Yes
Outgoing Nb of Patients	16	16	1	16.8	15.2	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Min Time	905	916	1.01	950.25	859.75	0.05	Yes
Max Time	1799	1811	1.01	1888.95	1709.05	0.05	Yes

(Registration)	Log file	Replay model	Ratio	Max range	Min range	Accepted error	Check
Incoming Nb of patients	104	104	1	109.2	98.8	0.05	Yes
Outgoing Nb of Patients	104	104	1	109.2	98.8	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Min Time	180	183	1.02	189	171	0.05	Yes
Max Time	361	361	1	379.05	342.95	0.05	Yes

Table 5.3: Validation results: replay simulation model vs log file

reality, depending on the hospital manager and/or the decision-maker. In fact, selecting this value needs investigations, studies, analysis, etc. Selecting this value is not within the scope of this research work.

From 02/01/2019 00:00 to 10/01/2019 20:26:21 (9 days)
The time in seconds

Hospital	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Incoming Nb of patients	962	942	0.98	1010.1	913.9	0.05	Yes
Outgoing Nb of Patients	962	942	0.98	1010.1	913.9	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Avg Length of Stay	3647.29	2945.73	0.81	3829.65	3464.93	0.05	No
Waiting Line	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Incoming Nb of patients	962	942	0.98	1010.1	913.9	0.05	Yes
Outgoing Nb of Patients	962	942	0.98	1010.1	913.9	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Max	11	15	1.36	11.55	10.45	0.05	No
Min	0	0		0	0	0.05	Yes
Avg Nb of patients	0.74	0.65	0.89	0.78	0.70	0.05	No
Avg Duration	588.16	666.33	1.13	617.57	558.75	0.05	No
Min Time	0	1.01		0	0	0.05	No
Max Time	2656.36	3857.84	1.45	2789.18	2523.54	0.05	No
Waiting Room	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Incoming Nb of patients	654	660	1.01	686.7	621.3	0.05	Yes
Outgoing Nb of Patients	654	660	1.01	686.7	621.3	0.05	Yes
Patients in Process	0	0		0	0	0.05	Yes
Max	13	11	0.85	13.65	12.35	0.05	No
Min	0	0		0	0	0.05	Yes
Min Time	121.45	130	1.07	127.52	115.38	0.05	No
Max Time	3592.68	3689	1.03	3772.31	3413.05	0.05	Yes
Registration	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Total Nb Of patients	962	941	0.98	988.05	893.95	0.05	Yes
Exam Room_1	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Total Nb Of patients	262	303	1.16	318.15	287.85	0.05	Yes
Exam Room_2	Replay model	Prospective model	Ratio	Max range	Min range	Accepted error	Check
Total Nb Of patients	159	210	1.32	220.5	199.5	0.05	Yes

Table 5.4: Validation results: prospective simulation model vs replay simulation model

5.4 Ramp-Up Phase

After validating the offline simulation models (replay and prospective) in Section 5.3.2, these models must be transformed into online models, which means connecting these models with the real world. To do this, the replay model will be transformed into a DTM and the prospective model will be transformed into a DTP, as discussed in Section 3.4.2. To make the connection, different sensors must be set up in the hospital to locate the different activities. According to the real patient pathway meta-model illustrated in Section 3.2, these sensors must be attached to each architectural item (room, open area, etc.), where the activities exist. In this work, virtual sensors were developed and installed inside the emulator. Figure 5.10 shows the locations of these sensors according to the architectural layout. Table 5.5 describes the purpose of each sensor in this layout. For example, sensor S1 will be used to detect the patient when he/she enters/leaves the hospital. In fact, each sensor is set up to detect two events; when the patient starts the activity and when he/she finishes the activity.

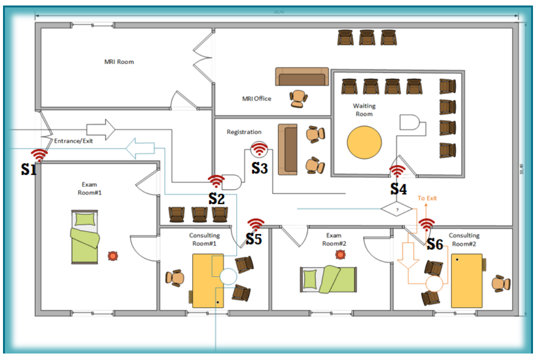


Figure 5.10: Architectural layout with sensors

In accordance with what has been discussed in Sections 3.4.2 (the technical aspects) and 4.4.4, we have developed a **virtual control component, which is called an event-handler component** to make the connection between the virtual world and the real world. This is

Sensor Number	Description
S1	Detects the patient when he/she enters/leaves the hospital.
S2	Detects the patient when he/she starts/finishes waiting in the waiting line.
S3	Detects the patient when he/she starts/finishes registration.
S4	Detects the patient when he/she starts/finishes waiting in the waiting room.
S5	Detects the patient when he/she starts/finishes the exam 1 activity.
S6	Detects the patient when he/she starts/finishes the exam 2 activity.

Table 5.5: Information related to the architectural layout

one of the requirements for making the connection with the real world, as depicted in the DT meta-model in Section 3.4.1.2. The main purpose of this component is to:

1. Receive events from the real world.
2. Execute actions according to the received events. In this work, four actions are executed by these components, as illustrated in algorithms 2 and 3 in Section 4.4.2 :
   (a) Create a virtual patient when the real patient enters the hospital.
   (b) Accelerate the virtual patient to the same location as the real patient, in case the real patient is faster than the virtual patient, or in case the virtual patient goes in a different direction (for example, if the real patient goes to activity called X, while the virtual patient goes to wrong activity Y).
   (c) Block the virtual patient from starting the activity until the real patient arrives at the same activity or a different activity. This is needed in case the virtual patient is faster than the real patient.
   (d) Remove the virtual patient from the model when the real patient leaves the hospital.

In order to convert a replay model into a DTM, virtual controls must be added to the replay model. This will allow the model to be connected to the real world. Rather than replaying the log file events, the model will reflect real-time events received from the real world. The replay model in this situation will be referred to as a DTM, as illustrated in Figure 5.11. Publish/Subscribe messaging can be used to exchange the data/information between the real world and the DTM.

Transforming the prospective model into a DTP means running the prospective model based on real-time data that was captured from the real-world sensors. In this case, the prospective model will be referred to as a DTP; as illustrated in Figure 5.12. The collected data will be used in two ways:

1. Update the static distribution that was discovered from the log file into dynamic empirical distributions, as discussed in Section 3.4.2
2. Before the model is run to anticipate the future, it must be fed with the current state of the hospital. In this work, the database is used to help in the initializing process for the DTP model.

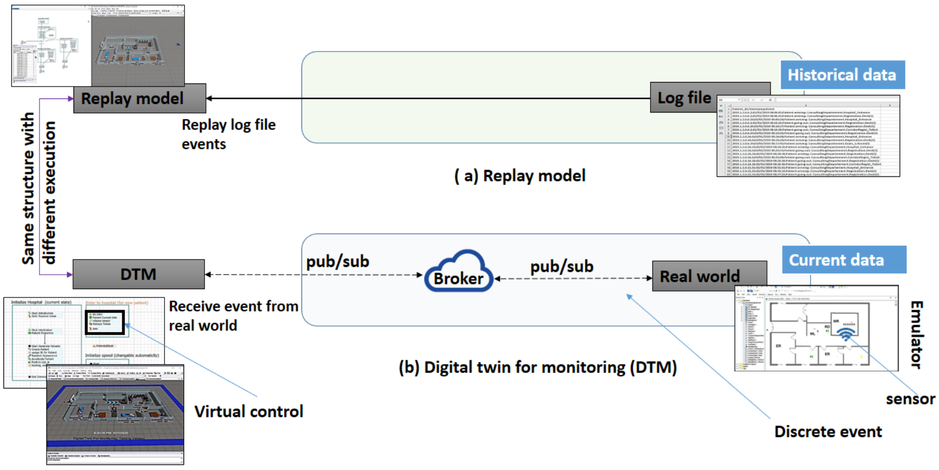


Figure 5.11: From replay model to DTM

Different replications can be made to predict the near future for more accurate prediction. The current state of the hospital will be used as input for all replications. However, because the model used some stochastic parameters, such as the pathway the patient may follow and the duration time for each activity, in addition to the different walking speeds for different patients, these parameters may lead to different replication results. In fact, more replications could provide us with more confidence in having less prediction error. “Confirmation comes from repetition ... Repetition is the basis for judging ... significance and confidence.” (Tukey, 1969)

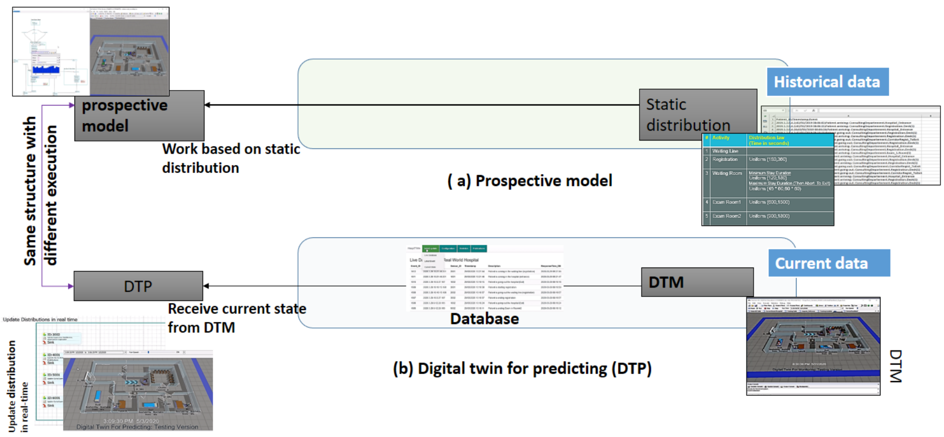


Figure 5.12: From prospective model to DTP

The transformation procedure that we used to convert the replay model into a DTM and the prospective model into a DTP is described in this section. Different sensors must be deployed in the real hospital to gather real-time data, and different virtual controls must be built within the virtual world models (DTM and DTP) to receive this data in order to complete this transformation. Because a real hospital was not available for this work, we opted to develop virtual sensors in the emulator to perform the same functions as the real-world sensors. For the virtual control, they will be created in the virtual model to receive data from different sensors regardless of the type of sensor (real or virtual). Figures 5.11 and 5.12 demonstrate the transformation process from replay model into DTM and from prospective model into DTP .

In conclusion, switching the connection from the emulator to the real world will not be a difficult process because the emulator and the real world have nearly the same behavior. In Section 5.5.1, we are going to illustrate how the DTM is connected to the real world (emulator, in our case), which means each time an event occurs in the real world, this event must be reflected in the DTM. Figure 5.13 depicts all of the technology used in this work (Witness for the emulator, FlexSim for the DT, MySQL as a database, publish/subscribe methods for exchanging data between the emulator and the DT, Open Platform Communications (OPC) as a communication protocol). In addition to the database used to initialize the DT models, the synchronization was done using publish/subscribe methods. PHP, Java, and FlexSim scripting languages were used to develop the needed algorithms.

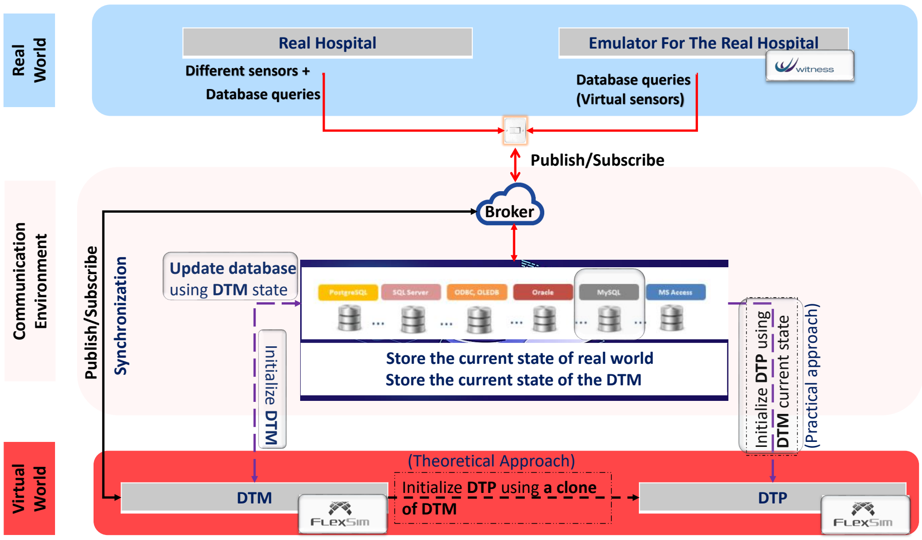


Figure 5.13: Connection between the real world and the virtual world

5.5 Running Phase

It Is considered as the last phase of our methodology, where the DTM and the DTP will be deployed in the real world. Sections 5.5.1, and 5.5.2 details how this deployment process will be done.

5.5.1 Digital twin for monitoring

In Section 5.4, we discussed how the replay model is transformed into the DTM and how the prospective model is transformed into the DTP. In this section, we are going to explain how the DTM has been deployed to monitor the real world (in our case, the emulator of the real world).

In Section 3.4.3, we discussed how the DTM will be connected for monitoring the real world. In fact, what was discussed before was more generic, which means the focus was not on specific technology, and the discussion was centered on the connection of the DTM and the real-world hospital. Due to the fact that this work does not include this type of hospital, the work in this section has been adapted for using an emulator and a certain type of technology, as depicted in Figure 5.13. Both the emulator and the real hospital have the same activities, and there are different sensors attached to the architectural items where these activities exist. This means that it does not matter if the data used to feed the DT have come from the real world (using real sensors) or from the emulator (using virtual sensors). In fact, from our point of view, the DTM must be blind to the source of the data. That is to say, the DTM needs to know the information that helps to perform the synchronization (e.g., location of each activity, the path of the detected patient, how to execute the action, etc.) regardless of where these data come from, as illustrated in Figure 5.13. For this work, Figure 5.7, shows the different pathways the patient may follow from one activity to another as well as it shows what are the activities that shall be connected to be synchronized with DTM.

To start the monitoring process, the DTM must be configured. First, there are some parameters in the DTM that must be set. For example, the different pathways that the virtual patient may follow after creation or after finishing an activity must be loaded into the DTM. In fact, the virtual patient does not know the next destination when he/she finishes the current activity. The reason behind this is that we use discrete events to monitor the real patient, as has been discussed in Section 4.3. Therefore, different pathways that have been discovered from the previous history must be loaded into the DTM. For example, when the patient is created at the entrance, the next destination the virtual patient may follow is the waiting line, and so on. Figure 5.14 shows a snapshot from our DTM where we load these pathways. Also, it is helpful to mention that these pathways could be updated or adjusted regularly based on the current data.

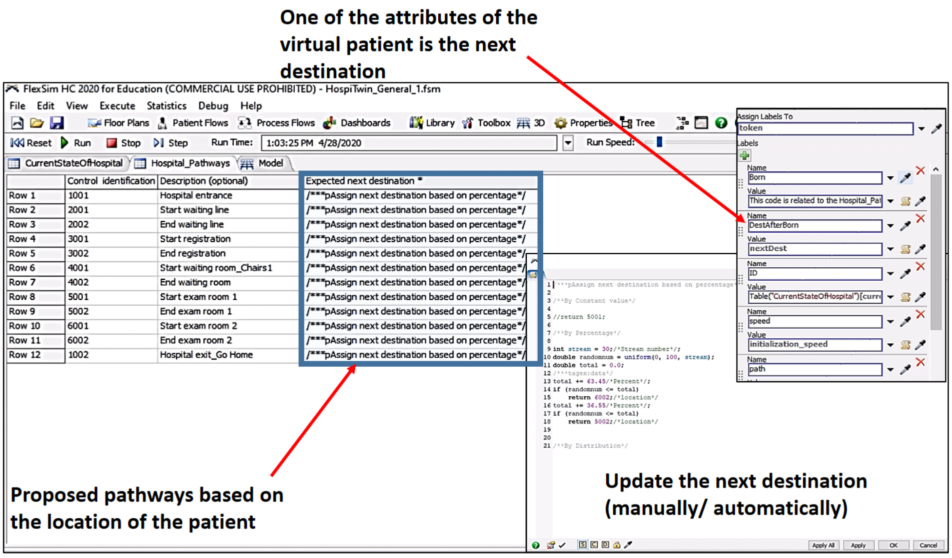


Figure 5.14: Loading proposed pathways in DTM

Another parameter that must be configured is the speed variable: in this work, three different speeds can be distinguished: (1) initialization speed, which is refers to the model, (2) teleportation speed, which refers to the speed of the virtual patient, and (3) custom speed, which also refers to the virtual patient:

1. Initialization speed refers to the speed used to initialize the DTM to the same state as the real world. Initialization speed can affect the initialization period, which is the time it takes to get the DTM to the same state as the real world. Actually, as the value of this speed increases, the initialization period will decrease (inverse proportion), as illustrated in Figure 5.15. Thus, the simulation model moves closer to reality. In other words, the value of this speed is very important. For example, suppose in the real world we have three patients in the waiting room; this means we need to have three patients in the chairs of the virtual waiting room at the time of running the DTM. If the initialization speed is slow, this means the initialization period will increase, and at the time of running the DTM, we can see that there is some delay before the patients sit on the chairs, but if the initialization speed is fast, that means the initialization period will be reduced, and at the time of running the DTM we will be able to see the virtual patients in the chairs directly without delay. This is related to the reason that there is a direct proportion between the initialization period and the period close to reality, as illustrated in Figure 5.15.

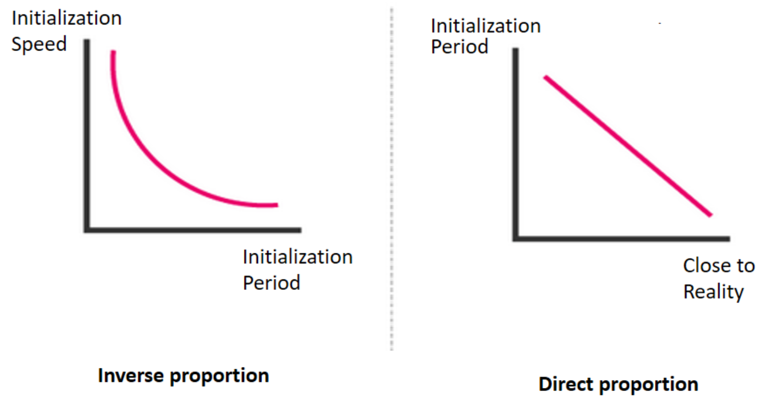


Figure 5.15: The relation between the initialization speed, initialization period, and being close to reality

2. Teleportation speed refers to the speed of the virtual patient. As stated in Section 4.4.2, there are three synchronization issues. One of these issues is that the real patient may be faster than the virtual patient. In this case, we need to accelerate the virtual patient to the same location as the real patient. To do this, we have already defined a parameter which is called a teleportation variable. This variable is used to store the teleportation speed. For example, the real patient arrives at the activity while the virtual patient is still walking. In this case, the speed of the virtual patient will be updated to have the value of the teleportation speed; this speed will move the patient to the same location as the real patient as soon as possible. For example, the teleportation speed could be 10 m/s.

3. Custom speed refers to the normal speed of the virtual patient, which corresponds to uniform(1,2), (Chandra et al., 2013) for the walking speed of pedestrians during walking and crossing. Normally, the virtual patient walks at this speed, except in the case of acceleration, where the speed will be changed to the teleportation speed.

All of the previous speed variables are dynamic and can be changed in a flexible way. Figure 5.16 shows the speed parameter variables for the developed DTM.

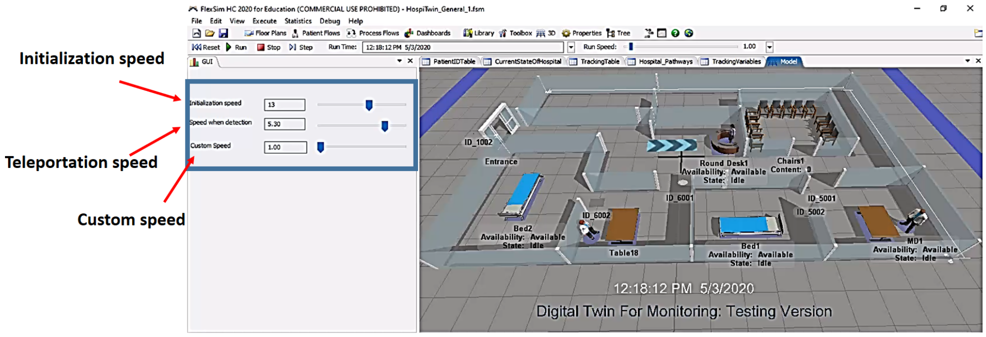


Figure 5.16: DTM different speed variables

The third parameter that must be configured is the location of the patient, which is a very important parameter. In this work we assign each patient with an attribute called the **born** attribute. At the time the patient is created in the model, this attribute stores the current location of the patient (e.g., waiting line, registration desk, etc). Based on this attribute, the DTM can determine the next destination for the patient. For example, the virtual patient will be created in the waiting room in the case where his/her corresponding real patient is in the waiting room. The next destination for the virtual patient could be an exam room or an exit from the hospital. This destination is determined because the DTM has some information about the location of the virtual patient, and based on the comparison of the location of the patient with historical data, the DTM can release the virtual patient to the expected next location (i.e., based on the historical data, the DTM knows, for example, that 70% of the patients will go to exam rooms after the waiting room, and 30% of the patients will go home). Furthermore, the value of the born attribute changes according to the location of the patient. For example, the current location of the patient may be the entrance door, so when the patient arrives in the waiting line, the value of the born attribute will be changed to the waiting line, and so on. Figure 5.17 illustrates the born attribute and the expected next location after the born attribute.

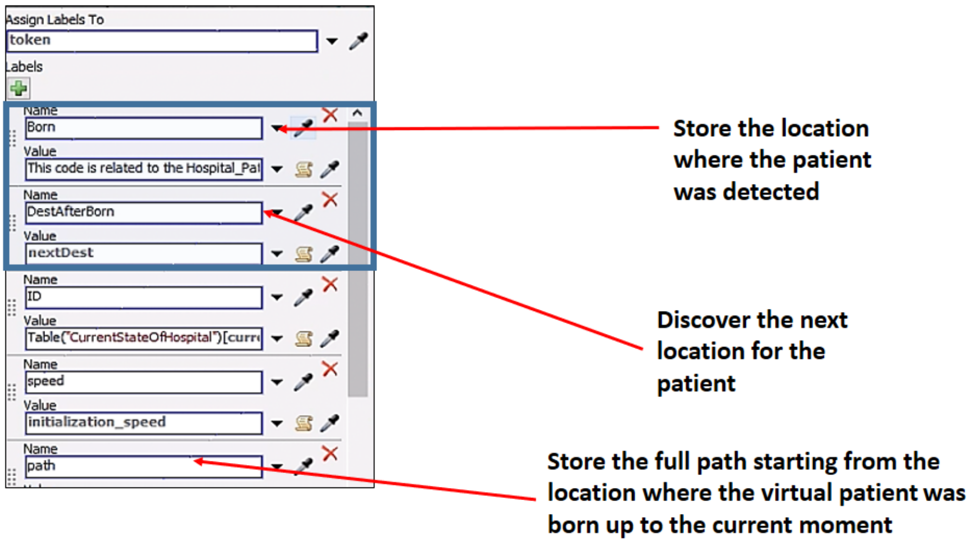


Figure 5.17: Current location and expected next location of the patient

Last but not least, to reflect reality, the simulation clock must be equivalent to the wall clock, which means the time scale between the real time and the virtual time must be 1:1 (1 second in the real world is simulated as 1 second in the simulation). In reality, the time scale of the simulation clock is greater than the time scale of the real world; for example, 10:1 (10 seconds in the real world is simulated in 1 second in the simulation). However, different algorithms can be used to solve this problem. Moreover, the DTM must be initialized before running with the same state as the current state of the emulator. To do this, we have used a private cloud database (IMT MINES ALBI servers). For example, at the time of running the DTM, we loaded the current state of the hospital from a database into the DTM. The value of the born attribute discussed previously was taken for each patient from the current state that is loaded from the database. For example, if the database mentions that the patient with ID: 12345 is in the waiting room, the born attribute for this patient will have a value called waiting room, and the next location could be any path, depending on the historical data. Figure 5.18 shows the current state of the hospital that will be used in the initialization process. When the DTM is started, these data will be loaded in real time (in a few microseconds) from the database.

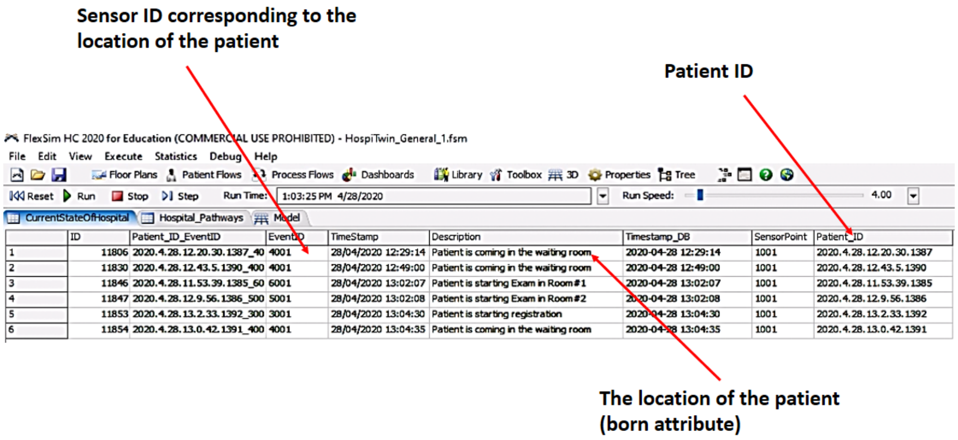


Figure 5.18: Current state of the hospital

Finally, the synchronization process between the DTM and the emulator will be done according to the steps that were discussed in Section 4.4.3. Figure 5.19 illustrates a snapshot of where the DTM is synchronized with the emulator. From this figure, the number of patients in the DTM can be seen as identical to the number of patients in the emulator. For example, both the DTM and the emulator have 8 patients in the waiting room and 1 patient in each exam room. In addition, the DTM clock is identical to the emulator clock, which is 6:30:36 pm, and the date is 5/31/2020.

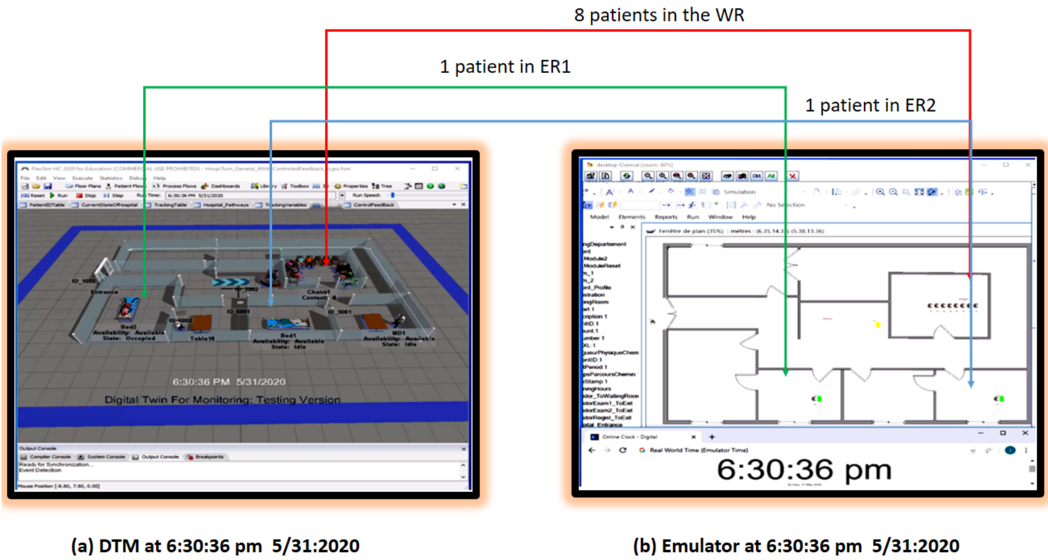


Figure 5.19: Experimental platform

Before summarizing this section, it is worth mentioning that a small-scale real experiment was carried out in the CGI department (Industrial Engineering Department) at IMT MINES ALBI in France. Radio Frequency Identification (RFID) readers were installed in different rooms, and each room represented an activity in a real hospital.

First, we were equipped with RFID tags. Then, we walked in front of the reader. Each time the RFID reader detected a tag, an event corresponding to this detection was published in the DTM. Then, the DTM created a virtual twin corresponding to the person with the tag in the model, and then this twin was allowed to move in any direction inside the model. When the real person changed rooms, the virtual twin in the DTM received another event for the new location of the real person. As a result, the virtual twin was accelerated to the same location as the real person to be synchronized with him/her. Figure 5.20 shows one of the real examples that was done to verify the model and check the technology that was used.

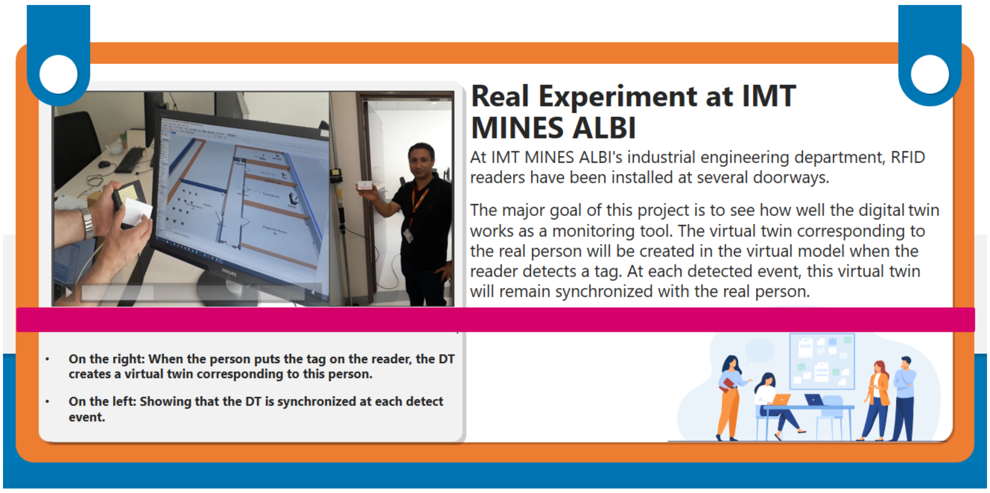


Figure 5.20: Real experiment at IMT MINES ALBI

In short, in connecting the DTM to monitor the real hospital (to synchronize the DTM with a real hospital) different steps had to be considered, including; (1) loading the different pathways the patient may follow, (2) configuring the different speeds (model speed, and patient speed), (3) initializing the model with current state of the hospital, and to do so, (4) initializing the simulation clock to be identical to the current world clock, and (5) synchronize the model.

To achieve the aforementioned steps, different attributes were assigned to each patient. They included: (1) the location where the patient existed (born attribute), (2) the next location where the patient had to go, and (3) different speed parameters to customize the patient's and the model's speeds. This section has demonstrated a proof of concept for the DTM by connecting this DT with an emulator. In Section 5.5.2, a prototype for the DTP will be shown.

5.5.2 Digital twin for predicting

A proof of concept for synchronizing the DTM with an emulator was demonstrated in Section 5.5.1. One of the key benefits of this synchronization is having near real-time information about the current state of the hospital (which is in the same state as the DTM). One of the key objectives for being in this stage is to anticipate the near future based on the hospital's current behavior. The current state of the real hospital from the DTM will be used as an input for the DTP. To accomplish this, the DTP will be run at a faster speed than the real-world clock to predict the future.

The DTP is a predictive DT with a similar structure to the DTM, except that it was built to anticipate the near future of the real world rather than to monitor it in real time. In fact, this DTP is a semi-online model, as it is connected to the real world only in real time to get the current state from the hospital as an input. Then it predicts the near future in an offline manner.

The DTP is used in this work to predict the near future using the dynamic empirical distribution indicated in Section 3.4.2. However, empirical distribution is only one mechanism that might be used to anticipate the future, and we have utilized only this mechanism due to lack of data; however, different mechanisms, such as artificial intelligence, could be employed for accurate prediction.

To run the predictive model, the DTP must first be initialized with the current state of the hospital. Two approaches can be used to initialize the predictive model: (1) taking a clone of the hospital's current state from the DTM and loading this clone inside the DTP: this is our theoretical approach, as illustrated in Figure 5.13. This approach was not successful because the commercial tool used in this work did not support it; (2) extract the current state of the hospital from the DTM into a database and then load these states from the database into the DTP; this is our practical approach, illustrated in Figure 5.13. In fact, We believe that the first approach is better than the second approach, because in the first approach there can be at least an approximate location for the patient in the hospital, while in the second approach, sometimes the location of the patient cannot be determined, especially if the patient exists between two events (when no sensor exists to detect the patient). Furthermore, knowing the specific position of the patient in the hospital will aid in better anticipating the patient's future. For example, if we know that a patient is at the registration desk, we can predict that this patient will be in the operating room in one hour, as depicted in Figure 5.21; (a) however, if we have incorrect information about the patient's location, the prediction result after one hour may indicate that the patient has left the hospital, as illustrated in Figure 5.21 (b) in the end, the second approach was employed in our work (initializing the prediction model from the DTP through database) due to the above-mentioned reason.

After initializing the DTP with the current state, the clock of the DTP must be synchronized with the real-world clock to be sure that the time and the state of the hospital are close to reality. In addition, the distribution that will be used to predict the future must be updated

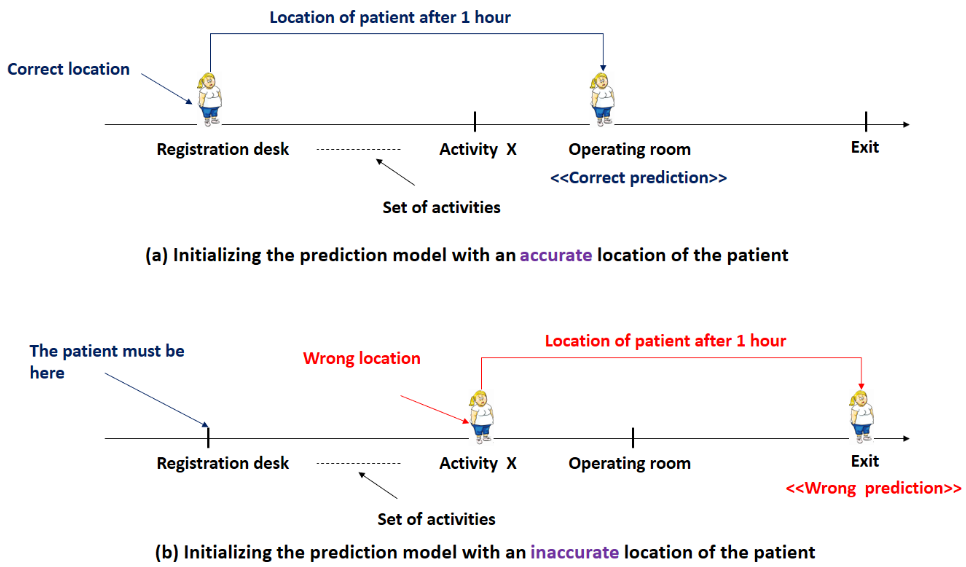


Figure 5.21: The current location of the patient provides more accurate prediction

just before running the DTP. Figure 5.22 shows a dashboard that was developed in this work to generate up-to-date distributions based on the data that was collected from the real world in real time. This dashboard was developed to show statistics related to the historical and current data collected from the real world (the emulator in this work).

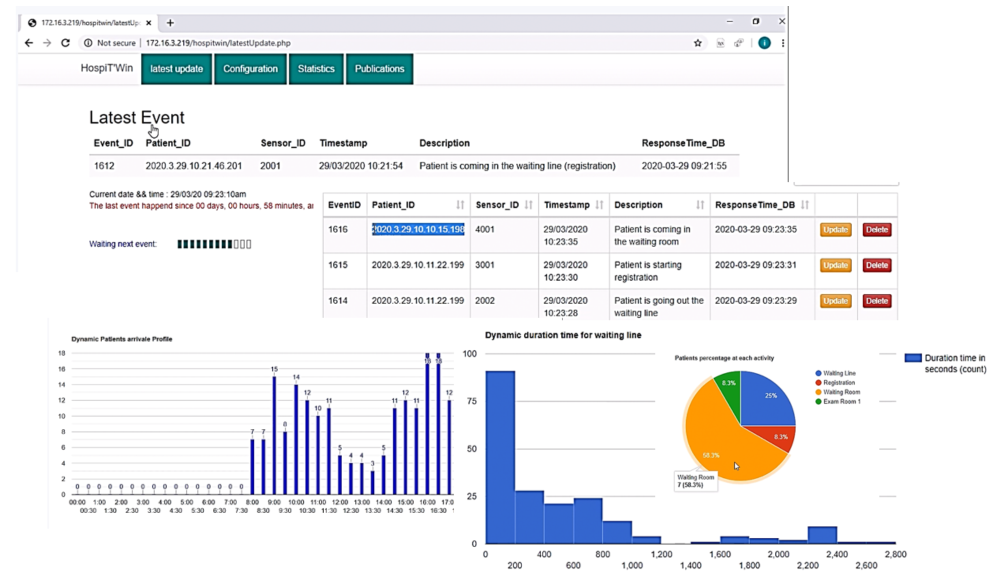


Figure 5.22: Samples from the dashboard

The last step is to run the DTP at a faster speed than the real world speed to anticipate the future. That means that the time scale of the DTP must be greater than the time scale of the emulator, as shown in Figure 5.23.

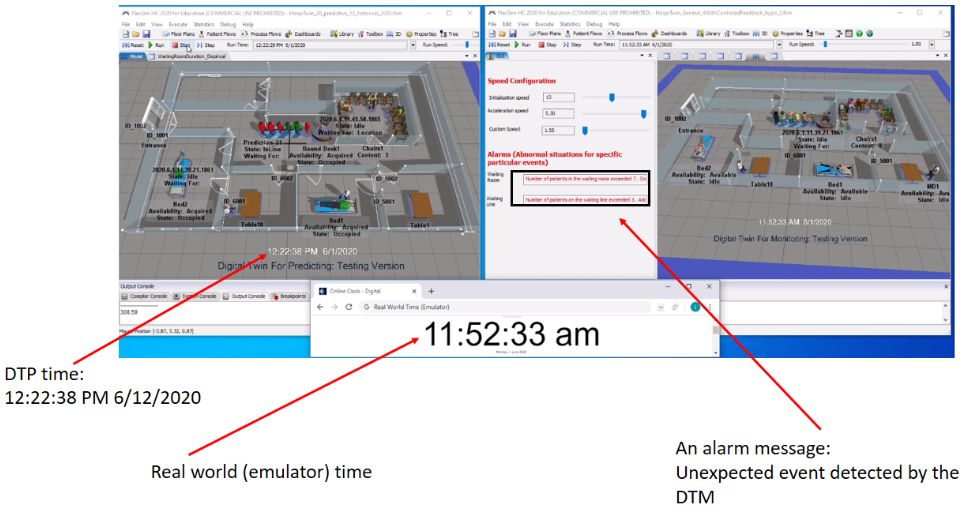


Figure 5.23: DTP time compared with emulator time

In Figure 5.23, the DTP runs faster than the real world, as can be seen from the clock. The current time in the real world is 11:52:33 am, whereas the time of the DTP is 12:22:38 pm; both of them have the same day, 6/12/2020.

For a quick run with our DTP, a proactive overview of the near future of the emulator (hospital) was chosen. Consequently, the DTP based on a proactive approach was used. A comparison between DTP outcomes and emulator outcomes in real time was performed. In other words, the events occurring in the DTP in the simulation time were compared with the events occurring in the emulator in real time. To do this, the DTP ran three replications per hour during a one-day experimental period (from 9:00 a.m. to 7:00 p.m. in real time) to quickly simulate a one-hour near future with the following steps:

1. Run the emulator.
2. Run the DTP at 9:00 am to quickly (in a few seconds) simulate and predict the next hour (10:00 am). Three replications were taken for each run. The average of these replications was calculated. Then, the statistics reports were collected at 10:00 am in the simulation time.
3. Compare the statistics that come from the emulator at 10:00 am in real time with the statistics that come from the DTP at the same simulation time.
4. Repeat steps 2 to 3 until 7:00 pm.

At 7:00 pm in real time, the mean absolute error was calculated to see the gap between the simulated information given by the predictive model every hour and the "real" information collected by the emulator at the same time. Figure 5.24 shows the results for four types of information (a blue line for real-world values and red line for DT values): (a) number of patients in the waiting line; (b) starting/ending events of registration activity; (c) number of patients in the waiting room; (d) starting/ending events of exam room activities. In Figure 5.24 (a, b, and d), the gap is small between the emulator and the predictive DT, while in Figure 5.24 (c) the gap is large. This could be due to several reasons: (1) there was not enough data used in this experiment to make the model more accurate; (2) at the initialization time, there may have been some patients trying to get out of the waiting room who had not yet arrived at the sensor, which means that for the DTP these patients were still in the waiting room, but for the emulator they had gone out of the waiting room (due to our practical approach); and (3) in this current work the focus was on building a general prototype for the DTP, where some factors were not considered, such as the amount of time the patient spends at each activity before running the DTP. This time was discussed in Section 4.3. However, all of the above-mentioned reasons could affect the statistics results.

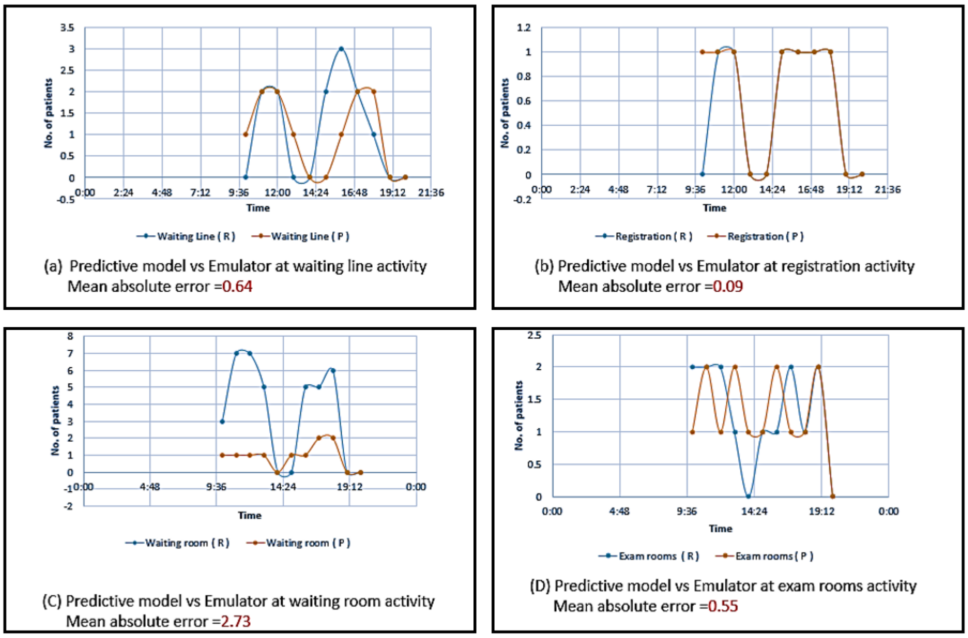


Figure 5.24: Statistics (DTP vs emulator)

In summary, the DTP was carried out by utilizing a proactive approach, as described in Section 3.4.3 (Running Phase). However, depending on the primary use, several approaches for using a DTP may be applied. A reactive approach, for example, may be used to run the DTP when an unexpected event occurs, whereas an on-demand approach could be used to run the DTP on demand for whatever reason. For instance, the manager might want to use the DTP to anticipate the behavior of one activity after an hour, and so on.

5.6 Conclusion

Basically, this chapter illustrates a proof of concept for the DTM, as well as a prototype for the DTP. A step-by-step methodology has been discussed in detail, from gathering data to running the DT models. To achieve the main goals for this chapter, an experimental platform that mimics a fictional hospital was developed.

The key models that must be constructed as a first step in designing a DTM and DTP have been shown in this chapter. This includes (1) a process flow model, (2) a replay model, and (3) a prospective model. The main benefits and applications of these models have been thoroughly discussed.

The transformation mechanism of offline simulation models to the DTM and the DTP has been detailed. The DTM, for example, must employ the following: (1) use real-time data, (2) initialize the DT with the hospital's current state, (3) set the simulation clock to the real-world time, and (4) at each detected event, synchronize the model with the real world. In addition, the DTP must employ the following: (1) instead of a static distribution, a dynamic empirical distribution has to be used, (2) the model must be initialized with the current state of the hospital before it can be executed, and (3) synchronize the simulation clock with the real-world clock, then accelerate the model to anticipate the future.

Different attributes have been assigned to the virtual patient as well as to the DTM model. This will help the DTM monitor the patient pathways in real time. This includes (1) different types of speed, such as teleportation and custom speeds assigned to the virtual patient, as well as the initialization speed assigned to the model itself, and (2) a proposed next-destination attribute has been assigned to the virtual patient. This will help the virtual patient determine the next path after finishing the current activity.

In short, some specific issues have been addressed: the initialization of DT models, real-time synchronization with the real world, and the connection between monitoring and prediction models. As a proof of concept, experiments have been carried out using an emulator of a hospital service connected to a DT built following our proposed methodology. Before connecting the DT with the real world, however, the different social, ethical, technical and security issues among others must first be resolved. All of these issues will be discussed in section 6.3.

6

Summary and Outlook

6.1 Research Objectives

A considerable amount of effort has gone into gaining thorough knowledge of a relatively new concept known as the Digital Twin (DT). The purpose of this research work was to investigate the most critical requirements and stages for designing and developing a proof of concept of a DT for patient pathways for the hospital of the future, called the HospiT'Win, as well as using Discrete Event Simulation (DES) to design this DT. The main objective of this DT is to be able to monitor the patient pathways in near real time whenever an event is detected in the real world, and this DT must be able to predict the near future of these pathways. To achieve the goal of this research, different research questions were developed, including:

1. How can DES be used to design a decision support tool based on a digital twin for real-time monitoring of patient pathways in the hospital and predicting their near future?

   This question was formulated in order to explore the different models needed to be constructed in building a DT of patient pathways for the hospital, as well as to investigate how to validate these models.

   The answer to this question was presented in Chapter 3. In short, a methodology for designing a DT was developed. This methodology is compromised of three phases and four steps. The phases of this methodology are (1) the design phase, (2) the ramp-up phase, and (3) the running phase. The steps of this methodology are (1) construction, (2) validation, (3) transformation, and (4) deployment.

   The construction step in the design phase shows the different models needed to construct this DT. These models are (1) the process flow model, (2) the replay model, and (3) the prospective model. Algorithms proposed for validating the DT, in addition to the indicators used for calibrating this twin, have been illustrated in the validation step.

2. How can the digital twin be enabled to have an accurate vision of the current situation of the patient pathways?

   This question was posed in order to achieve two goals. First, the initialization of the DT models. Second, synchronization of these models with the real patient pathways at each detected event. To reflect reality, before starting the Digital Twin for Monitoring (DTM) and the Digital Twin for Predicting (DTP), they must be initialized with the current state of the real patient pathways, and the simulation clock must be set to the same “wall clock" as in the real world. Following initialization, the DTM must be synchronized with real patient pathways. Each time there are changes in the real patient pathways, the DTM must reflect these changes in its model, whereas the DTP must be accelerated to be faster than the real patient pathways. The main reason behind this is that the DTM is used to monitor the real patient pathways, whereas the DTP is used to predict the future of the real patient pathways.

   The answer to this question was given in Chapter 4. In the initialization part, an initialization algorithm was proposed for initializing the DT models with the current state of the real patient pathways. Other proposed algorithms and GRAFNet charts were defined and used in the synchronization part to demonstrate the main concepts behind synchronizing the DTM of patient pathways with the real patient pathways at each event detected in the real patient pathways. However, different synchronization issues have been resolved, such as: (1) Accelerating the virtual patient to the same location as the real patient in case the real patient is faster than the virtual patient. (2) Blocking the virtual patient from starting the activity until the real patient arrives at the same activity. This can happen if the virtual patient is faster than the real patient. (3) Releasing the virtual patient to the same location as the real patient in case the two patients are going in different directions.

3. How can the Digital Twin for Monitoring (DTM) be triggered to run the Digital Twin for Predicting (DTP)?

   This question was undertaken in order to find out more information about the various approaches that can be utilized to switch from the DTM to the DTP. In fact, these two models are complementary to each other because the output of the DTM can be used as input for the DTP. For example, if an unexpected event is detected by the DTM, the current state from the DTM must be cloned and loaded inside the DTP, then the DTP will be run faster than the real-world clock to anticipate the future. In this work we have taken a clone for the current state from the database to initialize the predictive model, with the reason behind this being related to the technology we have used, as it did not support the cloning mechanism for the current state of one model to use with another model.

   The DTM has numerous advantages, including the ability to detect variance between the real and virtual worlds. Most of the time, we assume that it is the virtual patient who goes the incorrect way. On this basis, we synchronize the location of the virtual patient to the location of the real one. But then a question arises: “What will happen if the real patient, for whatever reason, moves in the wrong direction?” Indeed, one of the primary goals of this research is to improve patient pathways. If this problem occurs, we will need to conduct additional research in order to reduce similar cases or, in other words, to reduce the variances between the real and virtual worlds. Another advantage is that showing 3D patients in a 3D environment (the virtual hospital) is better than just showing the number of patients on a dashboard. The operations and the current location of the patients cannot be viewed on the dashboard, and even though we only monitored the patient pathways in this work, we feel it is worth mentioning. Furthermore, some patients will refuse to have their bodies monitored during surgery. Having an avatar that corresponds to this patient may make it more acceptable for the patient in this scenario. All of these benefits, as well as others, are contingent on the accuracy of the DT and the way of transforming the real world into a virtual (digital) world.

The answer to the question related to the switching between the DTM and the DTP can be found in Chapter 4. Different approaches have been illustrated for making the switch from the DTM to the DTP: a reactive approach, a proactive approach, and an on-demand approach. In a reactive approach, the DTP will be run at the time the DTM detects an unexpected event in the real world. This will help the decision-maker to make a decision on reducing the impact of the event in the near future of the hospital. In a proactive approach, the DTP will be run to anticipate a deviation that will occur in the future. This will help the decision-maker in reducing or preventing the occurrence of this deviation in the near future. Finally, in an on-demand approach, the DTP is run for whatever reason related to the issue the decision-maker wants to study or investigate.

For more clarification on the questions listed above, Chapter 5 has illustrated a proof of concept for a DTM and shown a prototype for a DTP. These illustrations were done using a near real-time connection between the DT models and an emulator that mimics a hospital. In this section, we have summarized the main objectives of this research work. In Section 6.2, we will now summarize the main outcomes and contributions that have been achieved.

6.2 Research Outcomes

In a complex environment such as a hospital, using a DT to monitor patient pathways and anticipate their near future is not easy, and it will take a significant amount of work to achieve this goal. In fact, as shown in Chapter 2, the literature review of this issue has shown that it has received little attention. In reality, most of the DT applications have been developed to serve the industrial domain. A few contributions has been made in the health care domain in general, but too few suggestions have been made for hospital management. For example, the contributions gleaned from the literature regarding hospitals focus only on how to improve one activity, such as the emergency department, or the intensive care unit. To the best of our knowledge, improving one activity in a complex network of activities could result in either a positive or a negative effect on the other activities. For example, improving the performance of activity X could create a bottleneck at activity Y, or it could improve activity Y. In our opinion, having a complete view of the whole process (patient pathways) is better than narrowly focusing on one activity, as illustrated in Section 3.3.

Along with the above-mentioned weaknesses, different limitations were discovered from reviewing the literature in Chapter 2 on designing a DT of patient pathways. For example, there are no rich process flow models that can be used to go directly from conceptual process models to online simulation models. In other words, the process flow charts that are present in the literature do not have enough knowledge to allow the DT designer and developer to design the DT. In fact, further knowledge is needed to enrich these process flow models, as illustrated in Section 2.3.3. Moreover, to the best of our knowledge, there are no meta-models that demonstrate how to design the DT of patient pathways. In addition, there is no single definition to clarify what the patient pathways in the hospital actually are. Most of the existing definitions are general definitions and do not focus on the patient pathways in the hospital, as illustrated in Section 2.2. Regardless of the DES's inability to construct this form of DT due to the different limitations illustrated in Section 2.3.3, there is no precise definition of what the DT of patient pathways is. All of these weaknesses have motivated us to come up with the following contributions that are summarized in Table 6.1:

Contribution	Description	Achievement	
Methodology for developing a DT for patient pathways inside the hospital.	This methodology aims to help in designing a discrete event simulation-based DT for real-time monitoring and near-future prediction of patient pathways in the hospital.	I. Life cycle to design the DT of patient pathways. II. Meta-model for real patient pathways in hospital. III. DT meta-model for patient pathways in hospital. IV. Process flow modeling notations.	Chapter 3
Mechanisms for initializing and synchronizing the DT of patient pathways with the real patient pathways.	The aim of these mechanisms are: I. To initialize the DT models (DTM and DTP) with the current state of real patient pathways. II. To synchronize the DTM to run in parallel with real patient pathways.	I. Algorithm for initializing the DTM and the DTP. II. Algorithms for synchronizing the DTM with the real patient pathways. III. GRAFNet chart to illustrate the different concepts behind synchronization.	Chapter 4
A proof of concept for the monitoring and predicting DT models.	This contribution aims to adopt and assess the suggested methodology as well as the developed initialization and synchronization algorithms to monitor the patient pathways and anticipate their near future.	I. Experimental platform. II. Dashboard. III. Digital twin for monitoring. IV. Digital twin for predicting.	Chapter 5

Table 6.1: Thesis contributions

This section has summarized the main contributions that were achieved in this thesis. In Section 6.3, we will illustrate the main limitations that will need to be overcome before the DT can be deployed in a real-world hospital.

6.3 Research Limitations

Testing the DT in an organizational context requires a hospital that deploys sensors to collect data about the patients, staff, etc. Furthermore, before adopting this DT in such a crucial health facility, a number of challenges must be addressed. These include:

1. Social issues: not all of the patients and the staff will agree to being monitored. In fact, most people desire to maintain independence, which is an inherently human trait (Percival et al., 2006). Moreover, despite the fact that monitoring a patient in a hospital will be used to improve his/her healthcare, some of the technology used will affect the patient's and the staff's privacy. In particular, some people fear the transmission of data and information between the different devices that will be used to build the DT, and between the autonomous/smart decision-making systems.

2. Privacy and security issues: Recently, this article appeared in the press: "A hospital in southwest France has seen some of its IT systems paralysed by a "ransomware" cyberattack, its management said Tuesday, the third such incident in the last month" (AFP, 2021).

   Sometimes it is difficult to find a way to keep patient data confidential, as there is a need to keep the services and the resources available for those who are authorized to use them. However, the DT needs to be protected from different types of attacks, such as internal attacks, which occur when an individual or a group within an organization seeks to disrupt operations or exploit organizational assets, or such as external attacks, which occur when somebody outside the organization attempts to exploit system vulnerabilities through the use of malicious software, hacking, sabotage or social engineering.

   From our point of view, the DT may lead to a new type of attack, which is a virtual attack, where an attacker can generate an attack using the virtual world to affect the real world. Here, attackers will have over-privilege access to the past data of the hospital, the current data, and the future data. To the best of our knowledge, no one has ever mentioned this type of attack, which could be considered one of the worst attacks that an organization could face. In fact, the attacker would not be able to just monitor the real-world or/and the virtual-world data. He/she would also be able to control the real-world environment and predict the future of this environment. Furthermore, the attacker could use the DT environment to simulate different types of attacks to discover vulnerabilities in the real world so that he/she could also generate physical attacks in addition to virtual attacks. Therefore, security is one of the most critical components that must be taken care of before applying the DT.

3. Ethical issues: the use of some of the technology, such as the Internet of Things (IoT) devices for monitoring the patient, can reduce visits from healthcare staff to the patient rooms. For example, the doctor will consider that his/her visit to the patient depends on the signals received from this technology. In addition, patients may change their behavior in response to the feedback from smart applications in order to conform with the IoT device's expectations.

   In the hospital, the IoT will be used to improve remote monitoring and develop quicker response times. Unfortunately, the nature of these technologies, which can be used simultaneously, can create opportunities for breaching personal or data privacy. Taking into account the consequentialist theory of ethics, this holds that "the morally right action is the one with the best overall consequences. (If there is no one best action because several actions are tied for best consequences, then of course any of those several actions would be right.)" (Haines, 2006). Because the user granted agreement to be observed, and the recorded data is analyzed for their own well-being, the usage of IoT in this context is ethical (Inieke, 2020). Hospitals are a vital aspect of everyday activities. Ethical aspects must be clarified and understood before applying the DT and its related technologies. In fact, the same technology could have positive and negative implications.

4. Technical issues: Hospitals may refuse to update their infrastructure and Internet network to include smart technology such as IoT devices and decision support systems (DT), even if they know that these technologies will help them. Sometimes the building structure and the hospital environment are not compatible with these technologies. This is the reason behind proposing this technology for the digital hospitals of the future.

5. Software issues: A tool that can clone the model with its current state from the DTM, and then use this clone to initialize the DTP with the same initialized state is needed. As far as we know, accurate initialization for the DTP may provide more accurate predictions for the future.

6. Uncertainty issues: Two types of events can be distinguished in this research work: (1) Unexpected events that were not planned and had never occurred before. For example, Coronavirus (COVID-19). (2) The second type can be referred to as unplanned but known events, which means that these events have occurred previously and that there are some potential solutions for dealing with this event if it occurs again. To handle this type of event, we recommend creating a predefined table in the DTM. This table will keep track of some of the acceptable Key Performance Indicators (KPIs) for each activity. If the DTM detects events but does not confirm the predefined KPI, the event is considered an unplanned but known event, and one of the acceptable historical solutions can be used to reduce or mitigate the impact of the event. For example, assume the maximum number of patients allowed at the registration desk is 5. This means that if the current number of patients at registration is 5, it is acceptable, but if there is a new patient coming to the registration desk, this leads to an alarm message being sent because the number of patients now exceeds the acceptable number (6>5). This event is considered an unplanned but known event. This event will be detected, and the solution may be to add a new clerk at registration.

One of the limitations is how to deal with unexpected events, especially those that may need to change in the model itself and/or the real-world hospital structure. A hospital must be aware that a backup of different kinds of resources is required such as fixed resources, human resources, etc. In case this event happens, the hospital must be recovered in a short time.

Sometimes, it might be difficult to tell the difference between unexpected and unplanned but known events. This is linked to the hospital's historical statistics. If the number of patients at the registration desk at a hospital exceeds five, for example, this event may be regarded as unexpected because, according to the hospital's history, the maximum number of patients who come to registration is three. For other hospitals, this event could be considered unplanned but known. This is related to the fact that these hospitals have experienced a similar event according to their past statistics. To consider this event an unexpected event in these hospitals, the number of patients must surpass fifteen at registration, and this could have happened in some crises or accidents, for example. As a result, the definition of an unexpected event must be clarified, and proactive solutions and plans must be considered to deal with such events.

6.4 Future Work

The DT of patient pathways is an exciting new area of research for which security will be one of its key challenges to address. Maintaining the security and privacy of this DT may improve the quality and effectiveness of patient care, including flexible elder and patient monitoring, drug management, patient experience enhancement, and cost reductions, among others. The most current and commonly used DTs contain vulnerabilities, posing risks to both its users and providers; thus, addressing the security and privacy of DT communications is very important. Moreover, accurate predictions followed by optimization are two key issues that must be addressed due to the complexity of each. In the future work, we intend to use AI and up-to-date data to address the first challenge, where AI seems to be capable of producing effective prediction models by using historical, contextual, and typically high-dimensional data. However, an optimization DT will be the third model that we will develop in our prospective work. The optimization model will benefit from the predictive results to optimize the patient pathways. For example, the predictive model will provide the prediction results for the state of the hospital in the near future and the optimization model will provide the best way to optimize the patient pathways according to the received outcomes from the predicting model. This section will illustrate a few details that describe our future work: cybersecurity, a DT for predicting and a DT for optimization.

6.4.1 Cybersecurity of the digital twin

Despite the benefits of the DT that allow the manager and the decision-maker to monitor, predict, control, and optimize the real-world environment, among other benefits, DTs could also be considered an open environment for attackers. If an attacker can access the DT, this means the attacker can monitor the physical environment that he/she wants to attack remotely. The attacker will also be able to predict the future of this environment, and thus will be able to define the date and a suitable time for generating the attack. In fact, the attacker can access and control all of the physical elements that can be accessed or controlled by the DT.

The accuracy of the DT can allow the manager and the decision-maker to make the right decision at the right time and in the right place. Imagine what would happen if the attacker attacked the communication network of the DT. In this case, the attacker could modify/alter the feedback from the DT for the decision-maker. In this case, the decision-maker would make the wrong decision because he/she had received misleading feedback. As well, the attacker could change the data coming from the real world to the DT. This would affect the accuracy of the DT.

Most of the attack operations are likely to take place virtually, which means the attacker could exploit the vulnerabilities of the virtual environment to access the real environment. These types of attacks can be called virtual attacks, and the attacker who generates this attack is a virtual attacker. In this situation, we can differentiate two types of virtual attacks:

- Active virtual attack: the attacker's efforts change or modify the data and the feedback transferred from/to the real world and virtual world. This attack could be dangerous because it could reduce the accuracy of the DT, as well as lead the decision-maker to make a wrong decision, which would affect the quality of the health care, the patient care, and patient satisfaction.
- Passive virtual attack: The attacker observes the data and the feedback, and the attacker can observe the future of the real environment (patient pathways) without generating any harm. In this case, the hospital data are no longer considered confidential.

Despite the fact that there is a variety of attacks that might exploit the DT to attack the real environment, the DT remains the key element in this environment. The DT can be instrumental in establishing a secure real-world environment, where the DT could be used to create a virtual copy of the Information Technology (IT) system and model a large number of attack tactics and techniques. The DT could be used to run hundreds of thousands of simulations to discover and find potential attack paths and potential intruders (hackers, worms or others) which could be utilized against the real system.

In conclusion, the DT may be a double-edged sword, meaning it could be used to secure the real-world environment against attacks, or could be used to attack the physical environment if it fell into the wrong hands. In this research we are interested in how to use and apply cybersecurity as a service for securing the DT. In other words, cybersecurity has to be implemented in the different layers (real world, virtual world, and the communication between them) with different mechanisms to secure the DT from attackers and keep the real environment secure. For example, in the real world, physical security must be implemented to secure the sensors, infrastructure, and the physical machines where the DT would be installed, etc. In the communication layer, network security must be implemented to ensure that the data is not spoofed or modified by a man-in-the-middle attack or any other types of attack. In addition, we need to be sure that the DT will not accept any connection requests other than the protected synchronization with the physical twin. In the virtual layer, we need to be sure that the models, algorithms and services are secure from vulnerabilities so a potential attacker would not be able to change the configuration or how these algorithms and models work. In the future work, we will be applying cybersecurity throughout the proposed three-layer architectures (real world, virtual world, and the communication environment between them).

6.4.2 DT for predicting

In this research, the DTP is used to predict the near future using the dynamic empirical distribution indicated in Section 3.4.2. However, empirical distribution is only one of the mechanisms that can be used to anticipate the future. We have utilized only this mechanism due to the lack of data, but different mechanisms, such as AI, could be employed for accurate prediction.

In this research work, the main focus is the DT for monitoring model. For us, this is a critical preliminary step. An accurate DTM must be a perfect shadow of the real world because the simulation model must be able to be cloned at any time and with any event to start a predictive simulation in a DTP. So if the initial state of the DTP does not fit with the current real state, and if the behavior of the DTP does not fit with the behavior of the real world, the prediction is certain to be far from what will really happen. Due to the lack of data for developing a proof of concept for a DTP, we decided to create a prototype for it.

We believe that an accurate and trusted DTP is essential for anticipating the future. In fact, the result from this twin will be used to make decisions corresponding to the real world. As a result, an accurate DTP leads to improving the quality of the health care. In future work, we could feed this DT with different types of data as well as using AI algorithms to improve the quality of the prediction.

6.4.3 DT for optimization

In this research work, we have focused on developing a proof of concept for a DTM and a prototype for a DTP. To complete our cycle, we believe that a Digital Twin for Optimization (DTO) is also required. First of all, when the DTM detects an unexpected event in the real world, the DTP must be run to anticipate the future by showing the effects of this event. If the future results from the predictive model do not fit with the objective of the decision-maker, the DTO can be used to minimize the deviation. It is able to take the future results from the DTP as inputs, and it will be able to run based on an optimization strategy (input values, objective function, etc.). After that, this DTO will provide the best possible combination of values for the decision-maker to consider in achieving his/her objectives. Figure 6.1 illustrates a complete cycle for HospiT'Win.

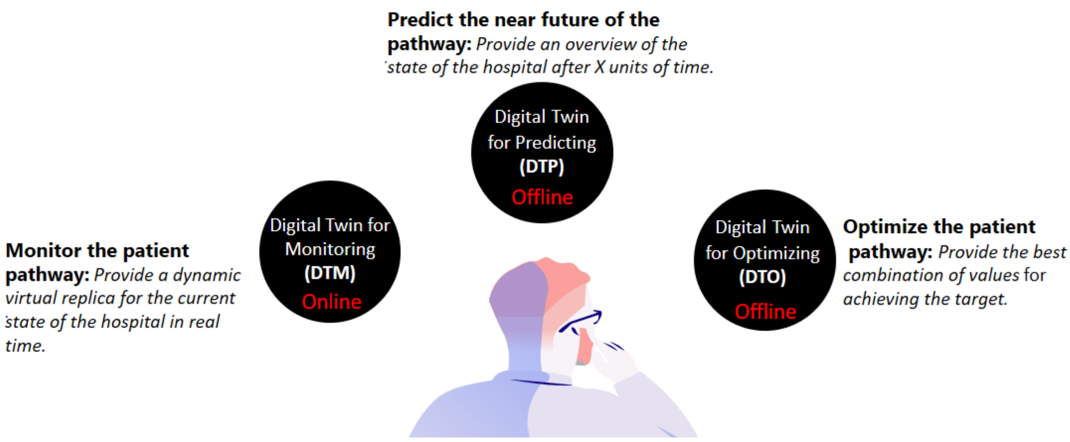


Figure 6.1: HospiT'Win: a combination of three digital twin models

Based on Figure 6.1, suppose the DTM detects twenty patients at the registration desk. This could be considered an unexpected event (overtaken threshold: send an alert if the number of patients at the registration desk is greater than 19). The decision-maker wants to see the impact of this event on the hospital after three hours. Based on the current state of the hospital, the decision-maker will run the DTP to anticipate the near future of the hospital in three hours. If the DTP model mentions that in three hours, all of the rooms in the hospital will be full, then the objective of the decision-maker to have no more than 80% of the rooms occupied will not be reached. As a result, the decision-maker will run the DTO based on the objective function (maximum occupancy of 80%) to determine the best combination of values for the decision variables for achieving this objective. The decision variables are, for example, the number of clerks at the registration desks, the duration time for operation X, the number of doctors at activity Y, the number of rooms to open, and so on. On this basis, the decision-maker will update the current values to the optimal values of the decision variables found by the DTO to achieve his/her objective.

6.5 Recommendations

The following recommendations list some of the basic elements that must be fulfilled for achieving the design and development of a DT for the patient pathways of the hospital.

- The DT must be initialized with enough accurate data to reflect the current state of the real patient pathways. To create a virtual patient pathway that is as close to the real one as possible, the real patient pathways must include as many sensors as possible. In fact, the data collected by sensors from the real patient pathways will be used in initializing and synchronizing the virtual patient pathways with real patient pathways. Fewer sensors will mean less data on the patient's location in real patient pathways. In other words, tracking the patient in areas where there are no sensors is difficult.

- The hospital's current and historical data must be used to create a predictive model. In other words, historical data alone may offer inaccurate prediction, especially if there is a long interval between gathering the data and running the predictive model, and current data alone will not allow accurate prediction models to be generated due to a lack of data. However, data must be accumulated continuously at each detected event. This will assist in covering as many cases as possible from the past, as well as cases that may not have been known in the past but have occurred today. In reality, combining previous and present data will yield more accurate predictive models that will provide more accurate outcomes.

- Security mechanisms must be considered in the real world, in the virtual world, and in the communication between them. Any vulnerability of these layers will provide a potential attacker with **control**. For example, if the attacker attacks the DT, this means the attacker would become the **real** owner of the real environment, where the attacker can generate his/her attack. Sometimes, attackers use brute-force [1] techniques to find the key to hacking perhaps just one vulnerable application. Imagine the attacker then gains control of the DT. This means the attacker can monitor the real world, predict its near future, and generate his/her attack based on his/her prediction, not only for that one application but also for all of the applications and machines controlled by the DT. In addition, in accessing the DT, the attacker can access all of the data stored in the DT database, which will violate the confidentiality of the facility, staff and patient data. Moreover, the attacker will be able to violate the availability of the system. For example, the authorized users of the DT would not be able to use it because the DT would be under the control of the attacker.

  The DT could impact the real world with positive or negative effects. The negative effects are as has been discussed previously, whereas the positive effects mean using the DT to secure the real world environment. For example, the DT could be used as a test environment to find the vulnerabilities in the real world. As a result, we can consider different mechanisms for protecting these vulnerabilities. Thus, these vulnerabilities will be protected from different types of attacks.

- Social, ethical, privacy and technical issues must be solved before applying the DT. For example, the staff and the patient must be informed of the importance of the DT and how the DT will increase the quality of health care as well as staff and patient satisfaction.

  The hospital's population may be frightened of the ability of the DT to collect large amounts of data on them. The point here is that we have to let them know that this

[1] A brute-force attack is a method of cracking passwords, login credentials, and encryption keys that relies on trial and error. It is a simple but effective method for gaining unauthorized access to individual accounts as well as systems and networks of businesses. In order to uncover the correct login information, the hacker will try various usernames and passwords, frequently utilizing a computer program to try a wide range of combinations.

data will be confidential data, and that no one can access it except those who are authorized. In this case, a privacy statement [2] should exist and be clear.

Different types of training must be scheduled to show the importance of the DT models (DTM and DTP). For example, to inform people that they are not being monitored simply for the reason of monitoring, or to disclose their data, or for hacking their privacy. The purpose of the DT is to improve the overall quality of health care, and it will be used in situations such as unexpected events (e.g., an emergency) because its use will provide the possibility of mitigating the effects of these events in the near future of the hospital. Other situations to improve could include, for example, reducing delays, increasing the satisfaction of patients and staff, managing the high flow of patients, dealing with elderly people, etc.

Last but not least, the infrastructure of the hospital must be ready for installing different IoT sensors. A high-fidelity environment is required. The IT team must be able to manage and configure the DT technologies. In addition, the hospital manager and the decision-maker must know how they can use this DT for making decisions related to managing/controlling and improving the patient pathways.

[2] In a privacy statement, the data controller should be identified, as well as the Data Protection Officer's contact information. It should also clarify why personal data is collected and processed, how it is used and disclosed, how long it is kept, and provide the controller's legal basis for processing.

Résumé étendu en français

Le domaine de la santé est l'un des secteurs qui croît le plus ces dernières années. (Bhat et al., 2014). D'après (World Bank, 2021), les dépenses mondiales liées à la santé représentaient environ 10% du produit intérieur brut (PIB) mondial en 2018. Ce sont les Etats-Unis qui y contribuent le plus par habitant, avec 10623,85 dollars américains (USD), suivis de la Suisse. Le Canada fait également partie des 20 premiers pays avec un total de 4994,90 USD investis par habitants, tandis que la moyenne mondiale est de 1111,082 USD (World Bank, 2021). De nos jours, les organismes de santé sont sous pression partout dans le monde, que ce soit dans les pays développés ou en développement. Ils ne permettent pas de répondre aux besoins des populations dont la durée de vie moyenne augmente chaque année et qui sont sujets à de plus en plus de maladies chroniques. De plus, les gouvernements définissent régulièrement de nouvelles directives de gestion de ces organismes imposant principalement de nouvelles contraintes en termes économique et de qualité des soins. C'est pourquoi les hopitaux, qui sont au coeur des activités de santé, peinent à faire face à ces évolutions dans un environnement de plus en plus soumis aux contraintes et à l'imprévisible. Par exemple, la planification des ressources dans un système aussi complexe qu'un hôpital est rendue de plus en plus difficile par la diversité des services, la croissance de la demande et la disponibilité variable des ressources selon les imprévus.

Les événements aléatoires et inattendus sont courants dans les organisations de santé, et plus particulièrement dans les hôpitaux. Les patients qui ne se présentent pas à des rendez-vous, les patients ou médecins qui sont en retard, une urgence nécessitant de repousser des interventions chirurgicales, un nombre insuffisant d'infirmiers ou un flux d'arrivée exceptionnel de patients suite à un accident de bus sont des exemples parmi d'autres d'aléas courants dans un hôpital. Les conséquences de ces aléas, accentuées par une capacité souvent insuffisante des ressources pour répondre à la demande, peuvent engendrer de longs temps d'attente pour les patients, à l'accueil et dans les salles d'attente. Parmi les imprévus mentionnés, certains sont dits internes car leurs causes sont internes à l'organisation, et d'autres sont externes car leurs causes sont extérieures à l'organisation, comme par exemple un accident sur l'autoroute dû à une mauvaise météo. Tous ces événements influent sur plusieurs facteurs : le bon fonctionnement de l'hôpital, la bonne prise en charge des patients, leur satisfaction, et la qualité des soins médicaux en général. A ce stade, il est important de noter que la qualité et la pertinence des décisions prises concernant ces aléas sont étroitement liées à la visibilité et à la connaissance qu'ont les managers sur la situation et son déroulement futur. Ainsi, le suivi en temps réel du parcours des patients (qui consiste en une séquence d'activités, cliniques, non-cliniques et administratives), est d'une importance cruciale pour informer les personnels médicaux et paramédicaux ainsi que les gestionnaires des hopitaux sur le bon déroulement de ces parcours. Ces informations aident à estimer et déterminer les ressources médicales et administratives à mettre en place pour prodiguer les soins adéquats à chaque patient, au bon moment et au bon endroit. (Huang et al., 2016).

Questions Métier

Ces travaux de recherche proposent de développer le principe du double numérique. Trois fonctions clés peuvent être définies et appliquées au secteur de la santé et plus particulièrement au suivi des parcours patients à l'hôpital. Cette section introduit les différentes questions spécifiques au métier du soin hospitalier, qui démontrent la pertinence de ces trois fonctions.

1. **Suivi des parcours en temps réel :** cette fonction permet d'avoir une vue en temps réel de l'activité de l'hopital et d'aider les décideurs à comprendre les difficultés des parcours patients actuels et la dépendance entre les différents services dans ces parcours. Ils pourront alors gérer les activités des patients de manière plus efficace. Enfin, le suivi en temps réel des parcours peut permettre de répondre à de nombreuses questions metiers qui se posent sur le terrain, comme par exemple:
    (a) Combien de patients se trouvent actuellement dans les locaux de l'hopital ?
    (b) Quelle est la localisation actuelle de chaque patient ?
    (c) Combien de temps chaque patient a-t-il passé en salle d'attente ?
    (d) Combien de temps chaque patient a-t-il passé avec le médecin ?
    (e) Quelle est la durée totale de séjour de chaque patient dans l'hôpital ? (LOS pour Length Of Stay) ?

2. **Prédiction du futur proche et évaluation de scénario spécifique :** Cela permet au manager de l'hôpital de réagir par anticipation face à des situations ou événements en cours. Avoir une vision de ce qu'il va arriver dans le futur est le meilleur moyen d'améliorer la situation présente. De plus, envisager une multitude de scénarios peut aider à évaluer la robustesse des stratégies de gestion face à des événements incertains. Ce type d'analyse peut également permettre de déterminer quelle stratégie est la plus adaptée et la plus résiliente face aux futurs changements. Enfin, prédire le futur aidera à répondre à diverses question métiers, concernant par exemple la gestion des aléas. Certaines de ces questions sont développées ci-dessous :
    (a) Que se passe-il en cas d'un afflux soudain et important de patients au service des urgences ?
    (b) Quel est l'impact d'une augmentation du nombre de patients sur les services de l'hôpital, son personnel, ses ressources, et autres ?
    (c) Que se passe t-il si un patient est absent à son rendez-vous médical ?
    (d) Que se passe t-il en cas de retard d'un médecin ?
    (e) Que se passe t-il si le nombre de patients dépasse la capacité de la salle d'attente ?
    (f) Comment évaluer l'impact d'un imprévu (urgence ou retard) en salle d'opération ?

3. **Amélioration de la qualité des soins médicaux :** l'idée de cette fonction est de collecter des informations sur différents soins médicaux appropriés et optimum dans des situations spécifiques et de les adapter dans des situations analogues. Elle permet aussi de répondre à différentes questions comme celles qui suivent afin d'améliorer la qualité et le bilan des soins de santé :
   (a) Comment gérer les ressources de l'hopital de manière efficace ?
   (b) Comment minimiser les coûts opérationnels ?
   (c) Comment réduire les temps d'attente des patients ou les retards dans les plannings ?
   (d) Comment maximiser la satisfaction des patients ?

Pour implémenter ces trois fonctions, nous pensons qu'un outil d'aide à la décision basé sur la technologie des doubles numériques (DN) est nécéssaire pour assister les décideurs d'un hôpital dans leur prise de décision en temps réel. Ces décisions seront donc basées non seulement sur la situation actuelle de l'hôpital et aussi sur la prédiction de son état dans un futur proche. En effet, la technologie des DN semble être une réponse pertinente aux besoins des managers. Ils consistent en des représentations virtuelles d'objets physiques réels et, dans notre cas, de processus réels, utilisés pour la compréhension, la prédiction et l'optimisation des flux de patients.

Ces dernières années, une vaste littérature s'est construite autour du concept de double numérique. Cet intérêt est principalement dû au potentiel que possèdent les DN pour : réduire les coûts des tests et vérifier les systèmes, produire des outils d'aide à la décision et d'alertes en temps réel, et prédire l'état futur de systèmes physiques (Madni et al., 2019). Bien que la terminologie ait changé au cours du temps, un DN peut être décrit comme la simulation d'un environnement **réel**, basée sur une étroite interaction entre les systèmes numériques et réels et utilisant un ensemble de modèles cohérents, détaillés et executables, ayant pour but d'aider aux processus décisionnels aux stades de conception ou d'exécution (Roland Rosen, 2018). Les objectifs d'un DN appliqué aux parcours de patients sont très ambitieux. La recherche scientifique sur les DN dans le domaine de la santé, contrairement au domaine industriel, est encore peu volumineuse, et nécéssite des approfondissements des concepts et des méthodologies et des exemples concrets d'application (Barricelli et al., 2019). Il n'existe pas de cadre methodologique pour structurer et guider la conception d'un DN en fonction des besoins d'un décideur en milieu hospitalier (Saracco, 2019). Actuellement, dans le domaine industriel, les DN s'appuient sur des solutions issues de l'internet des objet (IoT pour Internet of Things) et de l'intelligence artificielle (AI) pour effectuer des analyses approfondies de systèmes. Cela a pour but de trouver de nouveaux enjeux, de minimiser des temps d'attentes, de réduire les coûts de maintenance, de simuler des scénarios et d'adapter la production aux besoins de clients. Nos travaux de recherche ont pour but de montrer la pertinence de l'usage de DN dans la gestion d'un hôpital et les considérations pratiques qui en découlent. Ils contribuent à mettre en place les premières fondations d'un nouvel outil numérique d'aide à la décision appelé HospiT'Win, et proposent une méthodologie pour développer un DN afin de suivre et prédire les parcours patients en temps réel. Cette méthodologie inclut l'initialisation et la synchronisation du DN avec les états et événements du système réel, puis une prédiction dans un futur proche.

Etat de l'art

La première partie de l'état de l'art porte sur les défintions des parcours patients à l'hôpital Elles sont présentées dans le tableau 2.1. Il existe de nombreuses façons de considérer les parcours patients et les différentes définitions reflètent les différentes approches du parcours patient. Une première approche est, par exemple, de considérer les parcours patients uniquement selon les activités cliniques (les diagnostics, les traitements, les radios etc.).Une autre approche est de définir les parcours patients selon le type de maladie, chaque groupe de patients ayant une pathologie différente suivant un chemin différent. D'autres chercheurs ont défini les parcours patients comme un protocole composé de la séquence des activités cliniques à suivre. Enfin, d'autres chercheurs ont utilisé les parcours patients comme outil de coordination et de gestion de l'organisation des soins. C'est cette dernière définition qui est utilisée dans ce travail de recherche.

Depuis 1998, l'étude des parcours patients est considérée comme l'une des méthodes d'aide à la décision la plus efficace par les décideurs des hopitaux. Le suivi des parcours patients en temps réel permet de comprendre la complexité des parcours et les dépendances des services. Ainsi, on peut répondre à plusieurs questions de gestion telles que : (1) Combien de patients y a-t-il dans les locaux de l'hôpital ? (2) Quelle est la localisation actuelle de chaque patient ? etc. Cette recherche s'intéresse essentiellement aux parcours patients à l'hôpital. Pour cette raison, nous proposons cette définition des parcours patients à l'hôpital : un parcours patient est une séquence d'activités propre au patient qui commence à l'admission de celui-ci à l'hopital et qui finit lors de la sortie de cet hopital. Cette recherche s'intéresse au suivi de certaines activités propres aux parcours patients, telles que les activités d'attente, les activités administratives, les déplacements et les activités cliniques.

De nombreux outils ont été utilisés pour améliorer les parcours patients. Ils sont classés par (Aspland et al., 2019) en quatre catégories : (1) la modélisation stochastique, (2) l'exploration de données ou l'apprentissage automatique, (3) la simulation à événements discrets (SED), (4) l'optimisation et l'aproche heuristique. Ces outils peuvent être appliqués seuls ou combinés de manière hybride. Par exemple, la simulation peut s'utiliser seule, mais on peut aussi la combiner avec de l'apprentissage automatique. Cependant, leur utilisation est le plus souvent "hors ligne", ce qui signifie que les données utilisées ne représentent pas l'état du système en temps réel, mais un état antérieur. Malheureusement, les données historiques extraites de périodes passées peuvent, dans certains cas, ne plus refléter la réalité, ce qui conduit à des résultats non exploitables. Cette recherche propose d'étudier et de mettre en avant les bénéfices d'une utilisation "en ligne" de ces outils en général et de la simulation en particulier pour améliorer les parcours patients à l'hôpital. C'est le concept de base d'un double numérique. D'après l'état de l'art, il existe différentes définitions du DN comme illustré dans le tableau 2.3. Pour ce travail de recherche, le DN est un environnement réel simulé, basé sur de fortes interactions bidirectionnelles entre le monde virtuel et le monde réel. Il existe différents domaines d'application du DN, mais la plupart des travaux ont été effectués dans le domaine industriel. Peu de travaux ont été réalisés dans le domaine de la santé et peu d'entre-eux s'intéressent à la gestion des flux patients à l'hôpital en particulier. Le tableau 2.4 illustre différents exemples de projets de DN dans le secteur de la santé.

Ce tableau distingue trois principaux types de DN : un DN "produit" (par exemple pour le cœur du corps humain), un DN "opération" (par exemple pour une activité d'examen) et un DN "processus" (par exemple pour les parcours patients). Ces types de DN sont utilisés pour concevoir le système, surveiller son comportement actuel et prédire son avenir proche, entre autres. Nos travaux utilisent donc le concept de double numérique à base de simulation à événements discrets. En fait, la simulation à événements discrets (SED) est l'un des nombreux outils utilisés dans l'analyse et l'amélioration des services de santé (Kammoun et al., 2014) (Zhang, 2018). En effet, la SED est très intuitive et puissante pour analyser, évaluer et améliorer les systèmes de santé complexes (Jacobson et al., 2006). Il existe de nombreux domaines d'application en santé où la SED est utilisée. Par exemple, dans les chaînes d'approvisionnement des hôpitaux (Kammoun et al., 2014), les consultations externes (Al-Araidah et al., 2012), les services d'urgence (Connelly et al., 2004), les unités de soins intensifs (Z. Zhu et al., 2012), et d'autres encore. Plus de détails sur la SED sont développés dans la section 2.3.

Problèmes de Recherche

Le principal problème de la SED est qu'elle est très dépendante des données historiques (Hammad et al., 2011; Tavakoli et al., 2008). En effet, les modèles de simulation utilisent ces données sous la forme de statistiques et de distributions aléatoires, notament pour simuler les durées ou les quantités de façon réaliste. Mais cela peut entrainer des résultats faux ou imprécis. En effet, comme les données utilisées peuvent correspondre à des périodes passées qui ne sont plus représentatives du présent ou du futur proche, les résultats de la simulation risquent de ne pas correspondre à ce qui se déroule dans la réalité. C'est une problématique importante dans les systèmes dynamiques complexes comme ceux du domaine de la santé et en particulier dans les hôpitaux. De plus, ces simulations commencent dans un état initial **vide** et **inactif** (Hanisch et al., 2005), souvent très différent de l'état du système réel. Les rapports statistiques à la fin de l'éxécution de la simulation, en particulier si la durée est courte, peuvent donc être biaisés par cette différence de situation initiale. De plus, collecter, traiter, préparer des données pour alimenter le modèle de simulation est coûteux en temps, même lorsque les données sont disponibles dans un système d'information. En plus de ces inconvénients, travailler avec la SED nécéssite de l'éxpérience, un coût et du temps pour ajuster et calibrer le modèle. A l'hôpital, de nombreux paramètres ou facteurs sont en évolution constante, comme le nombre de patients, le nombre de demandes, les plannings du personnel médical, etc. Par conséquent, les modèles de SED traditionnels ne peuvent pas toujours traiter de la situation courante, et les prédictions auxquelles elles aboutissent sont peu fiables pour des analyses à court terme. Cette thèse utilise le concept de DN à base de SED avec des données mises à jour en temps réel pour répondre aux questions de recherche suivantes :

1. **Comment concevoir un outil d'aide à la décision à base de double numérique pour le suivi en temps réel des parcours patients à l'hôpital et la prédiction dans un futur proche ?**
   (a) Quels modèles doivent être utilisés pour construire un double numérique ?
   (b) Comment valider ces modèles ?
2. **Comment permettre au double numérique de fournir une vision cohérente de la situation réelle des parcours patients?**
   (a) Comment le DN peut-il être initialisé avec des informations actuelles correspondant à l'état réel des parcours patients ?
   (b) Comment les parcours virtuels des patients basés sur les événements discrets de la SED peuvent-ils être synchronisés avec les changements d'état des vrais parcours patients ?
3. **Quand utiliser le double numérique pour la supervision (Digital Twin for Monitoring (DTM) et quand utiliser le double numérique pour la prédiction, (Digital Twin for Predicting (DTP)) ?**
   (a) Dans quels cas/situations le modèle prédictif peut-il fonctionner (réactif, proactif ou sur demande) ?

Contributions de la Thèse

Dans ce travail de recherche, trois contributions ont été développées. Ces apports sont résumées à la suite.

Contribution n°1 : Méthodologie d'élaboration d'un double numérique pour les parcours patients à l'hôpital :

La méthodologie de conception d'un double numérique pour les parcours patients à l'hôpital à l'aide d'un SED est proposée dans ces travaux afin de répondre à la question de recherche n°1. La figure 3.1 présente le schéma général de cette méthodologie. Elle comporte trois phases et quatre étapes. Les phases sont les suivantes : conception, montée en puissance et exécution. Les étapes sont la construction, la validation, la transformation et le déploiement. Les deux premières étapes (la construction et la validation) se font hors ligne, sans relation avec la situation actuelle, et les données utilisées sont historiques. La deuxième étape (la transformation) se fait en ligne avec une connexion en temps réel à la situation actuelle. La troisième étape (le déploiement) se fait en ligne pour exécuter le Digital Twin for Monitoring (DTM) et (semi-online) pour exécuter le Digital Twin for Predicting (DTP), mais les deux DN sont initialisés avec les données actuelles. Dans la phase de conception, à l'étape de construction, deux types de données sont utilisées : (1) le plan archietctural de l'hôpital et (2) les données historiques du système d'information de l'hôpital (SIH) sous la forme d'un fichier log qui comporte les événements passés des parcours patients. Les données recueillies sont utilisées pour construire trois modèles de connaissance : un modèle de flux et deux modèles de simulation hors ligne appelés modèle "replay" et modèle prospectif, comme illustré sur la figure 3.5.

Le modèle de flux est considéré comme un modèle de connaissance et d'enrichissement qui fournit autant d'informations que possible sur les parcours patients, telles que les différents parcours, les différents types d'activités, la durée de ces activités, etc. Toutes ces informations sont synchronisées avec la réalité. Ce modèle de flux a été développé sur la base du méta-modèle DN des parcours patients présenté dans la section 3.4.1.2, et s'inspire des notations d'une norme industrielle japonaise. Ces notations sont illustrées dans le tableau 3.3.

Le modèle "replay" et le modèle prospectif sont des modèles de représentation virtuelle hors ligne, tous deux utilisés pour la validation du DN avant la connexion avec le monde réel. Ces deux modèles ont la même structure mais sont différents en terme de fonctionnement. Le modèle "replay" permet de rejouer les événements du fichier log extrait du système d'information hospitalier pour une période déterminée, tandis que le modèle prospectif simule le processus en utilisant des distributions de variables aléatoires (arrivées de patients, durée de chaque activité, vitesse de marche, règles de décision, etc.).

Dans l'étape de validation de cette même phase de conception, les modèles "replay" et prospectifs sont validés avant de les transformer en modèles en ligne (DTM et DTP). Cela signifie que le modèle "replay" doit être validé avant de le transformer en modèle DTM, de même que le modèle prospectif sera validé avant de le transformer en DTP s'exécutant sur la base de données actuelles et de distributions dynamiques. L'étape de validation fonctionne grâce à des indicateurs de performance (KPI) sélectionnés auxquels on attribue une valeur seuil. La figure 3.7 et la figure 3.8 résument le processus de transformation. Enfin, en phase de fonctionnement, les modèles de DN sont mis en œuvre dans un véritable hôpital. Lors de cette phase, les modèles de DN sont installés, configurés et personnalisés pour être utilisés dans l'hôpital en situation réelle. Au départ, les modèles de DN (DTM et DTP), sont installés sur les serveurs de l'hôpital, puis doivent être configurés en fonction des états et des comportements de l'hôpital. Le DTM doit notamment être initialisé avec l'état actuel de l'hôpital. Par exemple, si le véritable hôpital a un patient au bureau d'enregistrement, trois patients dans la salle d'attente, etc., le DTM doit être exactement dans le même état. Pour refléter la réalité, l'horloge du DTM doit être synchronisée avec le temps courant. Ensuite, le DTM doit être exécuté en parallèle du monde réel et en temps quasi réel. Par exemple, lorsqu'un événement se produit dans le monde réel, ce même événement doit se produire dans le DTM en temps réel. Si le DTM détecte un événement inattendu, comme un retard, un rendez-vous manqué, une saturation d'une salle d'attente, etc., le DTP peut être lancé pour prédire si cet événement affectera la réalité dans un futur proche. Après cela, une décision proactive pourra être prise pour tenter de réduire l'impact de cet événement.

Contribution n°2 : Mécanismes d'initialisation et de synchronisation du DN des parcours patients virtuels avec les parcours patients réels :

Des algorithmes d'initialisation et de synchronisation sont proposés pour répondre à la problématique soulevée par la deuxième question de recherche. Dans la phase d'initialisation, différents attributs, en plus de certaines fonctions de base, ont été proposés. Chaque patient virtuel dans le modèle DN est associé à un **ID**, ce qui permet à un patient virtuel unique dans le modèle DN de représenter un patient réel dans du monde réel. L'attribut **Born** représente l'emplacement actuel de ce patient dans le modèle DN en fonction de l'emplacement actuel du patient réel.L'attribut **emplacement suivant** a été ajouté au patient virtuel pour définir le prochain emplacement attendu où le patient doit se rendre.

Différentes fonctions sont proposées pour permettre l'initialisation du modèle. Par exemple, **Create (VirtualPatientID, Born)** pour créer des patients virtuels dans le DN en fonction des emplacements des vrais patients correspondant dans le monde réel, **AssignNextDistPath(VirtualPatientID)** pour attribuer un chemin à suivre aux patients virtuels, **InitializeClock(RealWorldClock)** pour initialiser l'horloge DN avec l'horloge du monde réel, etc. L'algorithme 1, section 4.3 illustre les principales fonctions du mécanisme d'initialisation.

Dans la phase de synchronisation, certaines fonctions et attributs supplémentaires ont été ajoutés au DN (DTM) en plus du contrôle virtuel qui est appelé gestionnaire d'événements pour résoudre trois conditions de synchronisation. Ces conditions peuvent être résumées par : (1) si le patient réel est plus rapide que le patient virtuel, alors le patient virtuel doit être accéléré jusqu'au même emplacement que le patient réel, (2) si le patient réel est plus lent que le patient virtuel, alors le patient virtuel arrive à une activité pendant que le patient réel marche. Dans ce cas, le patient virtuel doit être bloqué jusqu'à ce que le patient réel arrive au même endroit que le patient virtuel, (3) si le patient virtuel et le patient réel vont dans des directions différentes, alors le patient virtuel doit être déplacé au même endroit que le vrai patient. Pour résoudre ces conditions de synchronisation, un attribut **speed** a été ajouté au patient virtuel, en plus des autres fonctions utilisées pour réaliser la synchronisation telles que : **Accelerate(VirtualPatientID,NextLocation)**, **Block(VirtualPatientID)**, **Move(VirtualPatientID, NextLocation)**. Les algorithmes 2 et 3, section 4.4.2 illustrent les principales fonctions utilisées dans le mécanisme de synchronisation.

Des contrôles virtuels sont ajoutés au DN (DTM) pour permettre la synchronisation entre le monde virtuel et le monde réel. L'objectif principal de ces contrôles est de recevoir des événements du monde réel et d'exécuter des actions en fonction des événements reçus. Un langage de programmation GRAFNet est proposé pour montrer la logique qui est derrière ces contrôles ainsi que la logique pour réaliser le mécanisme de synchronisation. La section 4.4.1 traite de l'utilisation de GRAFNet plus en détail.

Contribution n°3 : Une preuve de concept pour le suivi et la prédiction des modèles DN :

Pour résoudre la troisième question de recherche, une plateforme expérimentale a été proposée comme banc d'essai pour tester les modèles de DN proposés (DTM et DTP). Cette plateforme se compose de deux éléments : (1) un émulateur d'un hôpital réel qui a été conçu et développé avec l'outil de simulation Witness, et (2) un DN qui a été conçu et développé avec l'outil de simulation FlexSim comme illustré sur la Figure 5.1. L'émulateur est conçu pour émuler un hôpital car il serait risqué de réaliser ces travaux de recherche en étant connecté à un "vrai" hôpital pour alimenter le DN. De plus, peu d'hôpitaux disposent aujourd'hui de capteurs permettant de suivre en temps réel les parcours patients. Même si ce type d'hôpital connecté est de plus en plus envisageable dans un avenir proche, nous pensons que l'utilisation d'une plateforme expérimentale reste pertinente car elle peut être utilisée en phase de test et de validation pour faciliter et garantir le déploiement réel. En effet, les tests et le processus de validation avec un émulateur se déroulent sans interrompre, affecter ou perturber les activités quotidiennes de l'hôpital. Après le test et le processus de validation, il sera simple de basculer la connexion en déconnectant le DN de l'émulateur et en le connectant au vrai hôpital, comme illustré sur la figure 5.1 (2).

Même si dans ce travail nous avons utilisé comme cas d'usage un hôpital fictif, celui_ci est représentatif de la réalité. Dans cet hôpital fictif, différents patients entrent à différents moments en suivant différents chemins pour recevoir un traitement. Ces patients marchent à des vitesses différentes. L'horloge de l'émulateur est réglée pour avancer en continu, comme une horloge du monde réel. De plus, différents capteurs virtuels ont été simulés pour collecter des données sur les parcours patients afin d'alimenter le DTM. Plus de détails sur cette plateforme expérimentale sont developpés dans la section 5.2.

La plateforme expérimentale est utilisée pour tester les trois approches différentes proposées dans ce travail de recherche :

1. Approche réactive : en cas d'un événement inattendu se produisant dans le monde réel, le décideur exécutera le DTP pour anticiper le futur proche et voir l'impact de cet événement sur l'hôpital réel, avant de chercher les solutions possibles pour réduire cet impact. Sur cette base, le décideur prendra une décision. La figure 3.11 illustre les principaux concepts derrière l'approche réactive.

2. Approche proactive : cette approche est considérée comme une approche périodique car le décideur exécute le DTP périodiquement, par exemple, toutes les **t** unités de temps pour anticiper le futur proche. Si le DTP détecte un écart dans le futur proche, le décideur essaiera de prendre une décision proactive pour réduire ou atténuer l'impact de cet écart avant qu'il ne se produise. La figure 3.12 illustre les principaux concepts se trouvant derrière l'approche proactive.

3. Approche à la demande : dans cette approche, le décideur peut choisir d'utiliser le DTP pour une raison quelconque. Par exemple, le décideur souhaite avoir un aperçu proactif d'une certaine activité ou un aperçu proactif de l'état de l'hôpital après trois heures. La figure 3.13 illustre les principaux concepts qui sous-tendent l'approche à la demande.

Conclusion

En raison de la dépendance des modèles traditionnels "Offline" de SED aux données historiques, et parce que ces données représentent des situations particulières du passé et ne sont pas forcément adaptées aux changements récents et présents, il peut être difficile de les utiliser pour prédire et anticiper avec précision les événements futurs, en particulier à court terme. Cette thèse cherche à explorer et prouver l'utilité des techniques de DN pour la gestion des parcours patients d'un hôpital. Elle met l'accent sur le processus de développement d'un DN basé sur la SED **en temps réel** pour éxécuter des modèles connectés et synchronisés au monde réel. Cette approche est dédiée **au suivi en temps réel des parcours patients et à la prédiction de leur futur proche**. Elle a pour but de gérer les comportements aléatoires, inhabituels et inattendus qui peuvent advenir dans un hôpital, et elle aide à la prise de meilleures décisions pour minimiser l'impact de ces situations imprévisibles. Différentes questions liées à la manière de développer, initialiser et synchroniser les DN sont discutées dans cette thèse. Nous résumons nos contributions dans le tableau ci-dessous. Les deux premières contributions sont scientifiques, tandis que la troisième est plus technique et démonstrative de la faisabilité du DN à base d'un SED. Elle consiste à implémenter concrètement notre méthodologie et permet ainsi de démontrer l'application pratique des deux premières contributions.

Contribution	Description	Résultats	
Une méthodologie pour développer un DN pour le parcours des patients en hôpital.	Cette méthodologie vise à faciliter la conception d'un DN à base d'un SED pour la supervision en temps réel et la prédiction des parcours de patient dans le futur proche dans l'hôpital.	I. Cycle de développement de conception des DN des parcours de patients. II. Meta-modèle des parcours réels de patients en hôpital. III. Meta-modèle des parcours des DN des patients en hôpital. IV. Modèles de flux de processus.	Chapitre 3
Des mécanismes pour initialiser et synchroniser les DN de parcours de patients avec les parcours réels des patients.	Les objectifs de ces mécanismes sont: I. Initialiser les modèles DN (DTM et DTP) avec l'état courant du parcours réel de patients. II. Aligner les DTM au même temps que les parcours réels de patients.	I. Algorithmes pour initialiser les DTM et les DTP. II. Algorithmes pour synchroniser les DTM avec les parcours réels de patients. III. Diagramme GRAFNet pour illustrer les concepts derrière la synchronisation.	Chapitre 4
Une démonstration de faisabilité des modèles DN de supervision et prédiction.	Cette contribution vise à adopter et évaluer la méthodologie proposée ainsi que les algorithmes d'initialisation et synchronisation développés pour superviser les parcours de patients et anticiper leur futur proche.	I. Plateforme expérimentale. II. Tableau de bord. III. Double numérique pour la supervision. IV. Double numérique pour la prédiction.	Chapitre 5

Limites de la recherche

Expérimenter un DN dans un contexte organisationnel requiert le déploiement de capteurs dans un hôpital pour recueillir des données sur les parcours patients, mais aussi éventuellement sur les localisations des personnels et des matériels mobiles. Ainsi, avant d'adopter un DN dans un établissement de santé, un certain nombre de problématiques doivent être étudiées, parmi lesquelles :

1. les problématiques sociales : tous les patients et personnels ne donneront pas leur accord pour être suivis et localisés en temps réel (voir aussi plus loin les problématiques d'éthique).

2. les problématiques de vie privée et de sécurité : il est parfois difficile de trouver un moyen de sécuriser les données confidentielles des patients afin que les services et les ressources ne soient accessibles qu'à ceux qui sont autorisés à les utiliser. Toutefois, les DN doivent être protégés de différents types d'attaques, comme les attaques internes, qui se produisent lorsqu'un individu ou un groupe au sein de l'organisation cherche à perturber son fonctionnement ou exploiter des actifs de l'organisation, ou des attaques externes, qui se produisent lorsqu'un individu ou un groupe externe à l'organisation

tente d'exploiter les vulnérabilités du système à travers des logiciels malveillants, des intrusions informatiques ou du sabotage.

3. les problématiques d'éthique : l'usage de certaines technologies, comme l'internet des objets (Internet of Things ou IoT) pour des équipements de suivi et de supervision des parcours patients, peut avoir un biais sur le comportement des soignants et des patients. Cela peut par exemple entrainer une réduction du nombre de visites du personnel médical dans les chambres de patients car le personnel peut considérer qu'il est surveillé et tracé par un signal émanant de ces technologies.

4. les problématiques techniques et économiques : Les hôpitaux peuvent ne pas être en capacité de mettre à jour leur infrastructure et leur réseau pour y inclure des technologies intelligentes comme des équipements connectés IoT et des systèmes d'aide à la décision à base de DN, même si ils savent que ces technologies pourraient leur être utiles.

5. les problématiques logicielles : Idéalement, le modèle de SED doit pouvoir être cloné à tout instant dans son état courant grâce au DTM. Ce clône doit pouvoir ensuite être utilisé pour initialiser le DTP dans un état qui est identique à celui du monde réel, avant de lancer une ou plusieurs simulations en acceléré pour se projeter dans le futur proche. Répondre à une telle problématique est un vrai défi technique car tous les outils de simulation ne permettent pas de réaliser à tout instant une sauvegarde du modèle et de son état pour pouvoir l'utiliser ensuite comme point de départ à une autre simumation en paralèlle. Dans ce travail de recherche, nous avons par exemple dû mettre en place un mécanisme d'initialisation du modèle à partir de données sauvegardées dans une base alimentée en temps réel.

6. les problématiques d'incertitude : deux types d'événements peuvent être distingués dans ces travaux de recherche : (1) Des événements imprévus qui ne sont jamais arrivés précédemment, comme par exemple la crise du Coronavirus (COVID-19). L'une des problématiques est de savoir comment réagir face à ces événements imprévus, particulièrement ceux qui nécéssiteraient une modification du modèle lui-même et/ou de la structure réelle de l'hôpital. Les décideurs dans un hôpital doivent être conscients que des remplacements de différents types de ressources doit être possibles, comme les ressources fixes, les ressources humaines, etc. Au cas où un tel événement se produit, l'hôpital doit pouvoir faire preuve de résilience afin de récupérer un fonctionnement accepatable en un minimum de temps. (2) Le deuxième type peut être décrit comme un événement non planifié mais connu, ce qui signifie que ces événements se sont déjà produits dans le passé et qu'il existe de potentielles solutions pour y faire face s'ils se produisent à nouveau. Pour gérer ce type d'événements, nous recommandons de créer un tableau prédéfini dans le DTM comportant un ensemble de valeurs seuils d'indicateurs de performance (KPIs pour Key Performance Indicators) à ne pas dépasser. Si le DTM détecte des événements qui ne satisfont pas les valeurs seuils des KPI définis précédemment, l'événement est considéré comme inattendu mais connu, et l'une des solutions peut être d'utiliser les données historiques pour réduire ou atténuer les impacts. Plus de détails sur ces limites de la recherche ont été discutés dans la section 6.3.

Organisation de cette thèse

Cette thèse est structurée de la façon suivante :

Chapitre 2 : Ce chapitre décrit les concepts généraux qui sont au coeur de la problématique de recherche, comme les parcours patients, les SED, et les DN. De plus, différents sujets explorés sont expliqués, comme les principaux composants des DN, leurs principaux usages, les domaines d'application, et les caractéristiques des sujets susmentionnés. Enfin, ce chapitre aborde les approches utilisées pour initialiser les modèles de DN et les synchroniser avec le monde réel. Les limites et les caractéristiques de ces approches sont également présentées.

Chapitre 3: L'objet de ce chapitre est d'introduire et discuter des méthodologies de recherche pour concevoir et développer les DN pour les parcours patients. Les concepts théoriques et technologiques nécessaires au développement des DN y sont illustrés. De plus, différentes techniques y sont développées, comme un méta-modèle pour le parcours réel des patients en hôpital, un méta-modèle d'un DN pour le parcours virtuel des patients en hôpital, et un flux de processus qui aide à concevoir et développer un DN pour les parcours patients en hôpital.

Chapitre 4 : Ce chapitre traite des différents concepts qui sont derrière l'initialisation et la synchronisation des modèles de DN pour les parcours patients. Certains algorithmes et extensions de langages de modélisation graphique, comme les réseaux de Petri et les GRAFCET, ont été fusionnés et utilisés pour clarifier et démontrer ces concepts. De plus, différents paramètres à prendre en compte pour l'initialisation des DTM et des DTP sont mis en relief. Diverses questions liées à l'initialisation et la synchronisation ont été résolues.

Chapitre 5: L'objet de ce chapitre est de poursuivre la méthodologie proposée dans le Chapitre 3 et d'utiliser les algorithmes d'initialisation et de synchronisation discutés dans le Chapitre 4 pour démontrer la faisabilité de notre approche grâce à la réalisation d'un prototype de DTM et de DTP à base d'un SED du commerce. Pour atteindre ce but, une plateforme expérimentale a été développée. Elle comporte un émulateur d'un hôpital réel qui est utilisé pour emettre des événements liés aux parcours patients et ainsi alimenter le DTM et le DTP pour les initialiser et si besoin les synchroniser.

Chapitre 6: Ce chapitre résume le principal objectif de ces travaux de recherche, ainsi que les questions de recherche et leurs réponses. De plus, ce chapitre explique les enjeux de recherche, ses limites, les travaux à venir, et des recommandations. En raison de l'importance de la sécurité des DN, certains points significatifs liés à la cybersécurité ont été illustrés dans les perspectives de ce chapitre.

List of Figures

List of Tables

Glossary

AI Artificial Intelligence . . . 2, 13, 40–43, 49, 142–144, 149

BSA Base Simulation Approach . . . 43–45

CPS Cyber-Physical System . . . 20, 23–25

DES Discrete Event Simulation . . . 4, 5, 7, 8, 14–19, 41, 43, 46, 47, 50, 53, 56, 58, 65, 79, 81, 82, 114, 119, 137, 139

DT Digital Twin . . . 2–8, 17, 19–30, 34, 35, 37–44, 46, 47, 49–54, 56, 58–61, 63, 65, 67–70, 72, 75–83, 85, 90, 94–96, 99, 100, 104, 109–111, 113, 114, 119, 123, 126, 132, 134–146, 159, 163

DTM Digital Twin for Monitoring . . . 5, 6, 26, 39, 47, 52, 53, 56, 58, 61, 65, 67–73, 75–77, 79–82, 90, 91, 94, 109–111, 113–115, 119, 120, 123, 124, 126–132, 135, 138, 139, 141, 143, 144, 146, 151–157, 163

DTO Digital Twin for Optimization . . . 53, 75, 144

DTP Digital Twin for Predicting . . . 5, 6, 39, 47, 52, 53, 56, 65, 67–70, 72–77, 79–82, 90, 91, 94, 110, 111, 113–115, 123, 124, 126, 132–135, 138, 139, 141, 143, 144, 146, 151–157

ENVA Essential Non-Value Added . . . 54, 58, 59

ER Exam Room . . . 61, 115

ER1 Exam Room 1 . . . 113, 114

ER2 Exam Room 2 . . . 113, 114

IoT Internet of Things . . . 2, 24, 29, 30, 34, 40, 41, 43, 49, 54, 56, 58, 67, 68, 83, 114, 141, 146, 149, 155, 156

IRTLS Indoor Real Time Location System . . . 56

KPI Key Performance Indicator . . . 15, 18, 66, 141, 156

LOS Length Of Stay . . . 3, 12, 148

ML Machine Learning . . . 13, 37, 41–43

N-VA Non-Value Added . . . 54, 58, 59

RD Registration Desk . . . 61, 70, 113–115

RFID Radio Frequency Identification 40, 43, 45, 56, 71, 131

RTLS Real Time Location System 40, 43, 60, 90

SCA State Collection Approach 43–45

VA Value Added 54, 58, 59

WL Waiting Line 15, 61, 113–115

WR Waiting Room 15, 70, 113–115

Bibliography

(Aasebø et al., 2012) U. Aasebø, H. H. Strøm and M. Postmyr. “The lean method as a clinical pathway facilitator in patients with lung cancer”. In: *The clinical respiratory journal* 6.3 (2012), pp. 169–174 (cit. on p. 12).

(AFP, 2021) AFP. *Third French Hospital Hit by Cyberattack | SecurityWeek.Com*. https://www.securityweek.com/third-french-hospital-hit-cyberattack. (Accessed on 08/12/2021). Mar. 2021 (cit. on p. 140).

(Aheleroff et al., 2021) S. Aheleroff, X. Xu, R. Y. Zhong and Y. Lu. “Digital twin as a service (DTaaS) in industry 4.0: an architecture reference model”. In: *Advanced engineering informatics* 47 (2021), p. 101225 (cit. on pp. 34, 35).

(Ahuett-Garza et al., 2018) H. Ahuett-Garza and T. Kurfess. “A brief discussion on the trends of habilitating technologies for Industry 4.0 and Smart manufacturing”. In: *Manufacturing letters* 15 (2018), pp. 60–63 (cit. on p. 43).

(Alam et al., 2017) K. M. Alam and A. El Saddik. “C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems”. In: *IEEE access* 5 (2017), pp. 2050–2062 (cit. on p. 20).

(Alizadehsalehi et al., 2020) S. Alizadehsalehi, A. Hadavi and J. C. Huang. “From BIM to extended reality in AEC industry”. In: *Automation in construction* 116 (2020), p. 103254 (cit. on p. 42).

(Anderson et al., 2004) D. M. Anderson and A. E. Michelle. “Mosby’s medical, nursing, & Allied Health Dictionary, sesta edizione”. In: *New York, Piccin* (2004) (cit. on p. 10).

(Arafsha et al., 2019) F. Arafsha, F. Laamarti and A. El Saddik. “Cyber-physical system framework for measurement and analysis of physical activities”. In: *Electronics* 8.2 (2019), p. 248 (cit. on pp. 24, 25).

(Araghi et al., 2018) S. N. Araghi, F. Fontanili, E. Lamine, L. Tancerel and F. Benaben. “Monitoring and analyzing patients’ pathways by the application of Process Mining, SPC, and I-RTLS”. In: *IFAC-PapersOnLine* 51.11 (2018), pp. 980–985 (cit. on p. 2).

(Al-Araidah et al., 2012) O. Al-Araidah, A. Boran and A. Wahsheh. “Reducing delay in healthcare delivery at outpatients clinics using discrete event simulation”. In: *International journal of simulation modelling* 11.4 (2012), pp. 185–195 (cit. on pp. 4, 150).

(Aspland et al., 2019) E. Aspland, D. Gartner and P. Harper. “Clinical pathway modelling: a literature review”. In: *Health systems* (2019), pp. 1–23 (cit. on pp. 12, 13).

(Augusto et al., 2018) V. Augusto, M. Murgier and A. Viallon. “A modelling and simulation framework for intelligent control of emergency units in the case of major crisis”. In: *2018 winter simulation conference (WSC)*. IEEE. 2018, pp. 2495–2506 (cit. on pp. 23, 24).

(Autiosalo et al., 2019) J. Autiosalo, J. Vepsäläinen, R. Viitala and K. Tammi. “A feature-based framework for structuring industrial digital twins”. In: *IEEE access* 8 (2019), pp. 1193–1208 (cit. on p. 43).

(Ayache, 2019) N. Ayache. “AI and healthcare: towards a digital twin?” In: *MCA 2019 - 5th international symposium on multidiscplinary computational anatomy*. Fukuoka, Japan, Mar. 2019 (cit. on pp. 22, 24).

(Ayache et al., 2011) N. Ayache, O. Clatz, H. Delingette, G. Malandain, X. Pennec and M. Sermesant. “Towards a personalized digital patient for diagnosis and therapy guided by image”. fre. In: *Medecine sciences: m/s* 27.2 (Feb. 2011), pp. 208–213 (cit. on pp. 22, 24).

(Aydt et al., 2008) H. Aydt, S. Turner, W. Cai and M. Y. H. Low. “An agent-based generic framework for symbiotic simulation systems”. In: *Multi-agent systems: simulation and applications* (2008) (cit. on p. 44).

(Aydt et al., 2009) H. Aydt, S. J. Turner, W. Cai and M. Y. H. Low. “Research issues in symbiotic simulation”. In: *Proceedings of the 2009 winter simulation conference (WSC)*. IEEE. 2009, pp. 1213–1222 (cit. on pp. 25, 26).

(Baillargeon et al., 2014) B. Baillargeon, N. Rebelo, D. D. Fox, R. L. Taylor and E. Kuhl. “The living heart project: a robust and integrative simulator for human heart function”. en. In: *European journal of mechanics - a/solids*. Frontiers in Finite-Deformation Electromechanics 48 (Nov. 2014), pp. 38–47 (cit. on pp. 22, 24).

(Banks, 2005) J. Banks. *Discrete event system simulation*. Pearson education India, 2005 (cit. on p. 14).

(BANKS et al., 2010) J. BANKS and J. CARSON. “S.; NELSON, Barry. L.; NICOL, David M”. In: *Discrete-event system simulation. 5nd ed. New Jersey: prentice hall* (2010) (cit. on p. 14).

(Barricelli et al., 2019) B. R. Barricelli, E. Casiraghi and D. Fogli. “A survey on digital twin: definitions, characteristics, applications, and design implications”. In: *IEEE access* 7 (2019), pp. 167653–167671 (cit. on pp. 2, 23, 38, 149).

(Barricelli et al., 2020) B. R. Barricelli, E. Casiraghi, J. Gliozzo, A. Petrini and S. Valtolina. “Human digital twin for fitness management”. In: *IEEE access* 8 (2020), pp. 26637–26664 (cit. on pp. 23, 24).

(Bashir, 2010) G. Bashir. “Technology and medicine: the evolution of virtual reality simulation in laparoscopic training”. In: *Medical teacher* 32.7 (2010), pp. 558–561 (cit. on p. 41).

(Basketter et al., 2018) V. Basketter, M. Benney, L. Causer, L. Fleure, D. Hames, S. Jones, K. Patel and L. White. “Continuous, comprehensive and crucial care: The role of the CNS in the metastatic castration-resistant prostate cancer patient pathway”. In: *British journal of nursing* 27.4 (2018), S1–S8 (cit. on p. 8).

(Basu et al., 2021) T. Basu, O. Bannova and J. D. Camba. “Mixed reality architecture in space habitats”. In: *Acta astronautica* 178 (2021), pp. 548–555 (cit. on p. 42).

(Bazilevs et al., 2015) Y. Bazilevs, X. Deng, A. Korobenko, F. L. di Scalea, M. Todd and S. Taylor. “Isogeometric fatigue damage prediction in large-scale composite structures driven by dynamic sensor data”. In: *Journal of applied mechanics* 82.9 (2015), p. 091008 (cit. on p. 21).

(Bernardi et al., 2019) S. Bernardi, C. Mahulea and J. Albareda. “Toward a decision support system for the clinical pathways assessment”. In: *Discrete event dynamic systems* 29.1 (2019), pp. 91–125 (cit. on p. 10).

(Bevilacqua et al., 2020) M. Bevilacqua, E. Bottani, F. E. Ciarapica, F. Costantino, L. Di Donato, A. Ferraro, G. Mazzuto, A. Monteriù, G. Nardini, M. Ortenzi et al. “Digital twin reference model development to prevent operators’ risk in process plants”. In: *Sustainability* 12.3 (2020), p. 1088 (cit. on pp. 34, 35).

(Bhat et al., 2014) S. Bhat, E. Gijo and N. Jnanesh. “Application of lean six sigma methodology in the registration process of a hospital”. In: *International journal of productivity and performance management* (2014) (cit. on pp. 1, 147).

(Bielefeldt et al., 2015) B. Bielefeldt, J. Hochhalter and D. Hartl. “Computationally efficient analysis of SMA sensory particles embedded in complex aerostructures using a substructure approach”. In: *ASME 2015 conference on smart materials, adaptive structures and intelligent systems*. American society of mechanical engineers. 2015, V001T02A007–V001T02A007 (cit. on p. 21).

(Bowers, 2009) B. Bowers. “The role of community nurses in providing a seamless patient pathway”. In: *British journal of nursing* 18.16 (2009), pp. 1010–1013 (cit. on p. 8).

(Braun et al., 2014) R. Braun, H. Schlieter, M. Burwitz and W. Esswein. “BPMN4CP: design and implementation of a BPMN extension for clinical pathways”. In: *2014 IEEE international conference on bioinformatics and biomedicine (BIBM)*. IEEE. 2014, pp. 9–16 (cit. on p. 13).

(Burwitz et al., 2013) M. Burwitz, H. Schlieter and W. Esswein. “Modeling clinical pathways-design and application of a domain-specific modeling language”. In: *Wirtschaftsinformatik proceedings* (2013), pp. 1325–1339 (cit. on p. 13).

(Cardin et al., 2011) O. Cardin and P. Castagna. “Proactive production activity control by online simulation”. In: *International Journal of Simulation and Process Modelling* 6.3 (2011), pp. 177–186 (cit. on p. 46).

(Carson et al., 1997) Y. Carson and A. Maria. “Simulation optimization: methods and applications”. In: *Proceedings of the 29th conference on Winter simulation*. 1997, pp. 118–126 (cit. on p. 75).

(Chan et al., 2019) K. A. Chan, M. J. J. Pan, B. T. Chua, X. M. Hu and M. Y. H. Low. “Resources optimisation in new hospital central kitchen design–a discrete event simulation approach”. In: *Asian simulation conference*. Springer. 2019, pp. 251–259 (cit. on pp. 16, 17).

(Chandra et al., 2013) S. Chandra and A. K. Bharti. “Speed distribution curves for pedestrians during walking and crossing”. In: *Procedia-social and behavioral sciences* 104 (2013), pp. 660–667 (cit. on p. 128).

(Cheng et al., 2016) R. Cheng, C. Macal, B. Nelson, M. Rabe, C. Currie, J. Fowler and L. H. Lee. “Simulation: the past 10 years and the next 10 years”. In: *2016 winter simulation conference (WSC)*. IEEE. 2016, pp. 2180–2192 (cit. on p. 17).

(Cimellaro et al., 2017) G. P. Cimellaro, M. Malavisi and S. Mahin. “Using discrete event simulation models to evaluate resilience of an emergency department”. In: *Journal of earthquake engineering* 21.2 (2017), pp. 203–226 (cit. on pp. 15, 17).

(commission et al., 2002) I. electrotechnical commission et al. “Grafcet specification language for sequential function charts”. In: *International Standard, IEC* 60848 (2002), p. 94 (cit. on pp. 83, 84).

(Connelly et al., 2004) L. G. Connelly and A. E. Bair. “Discrete event simulation of emergency department activity: a platform for system-level operations research”. In: *Academic emergency medicine* 11.11 (2004), pp. 1177–1185 (cit. on pp. 4, 150).

(Corless et al., 2018) L. Corless and I. Brew. “Patient pathway: the ideal approach”. In: *British journal of nursing* 27.Sup3 (2018), S14–S19 (cit. on p. 8).

(Cubukcuoglu et al., 2020) C. Cubukcuoglu, P. Nourian, I. S. Sariyildiz and M. F. Tasgetiren. “A discrete event simulation procedure for validating programs of requirements: The case of hospital space planning”. In: *SoftwareX* 12 (2020), p. 100539 (cit. on pp. 16, 17).

(Danilczyk et al., 2019) W. Danilczyk, Y. Sun and H. He. “Angel: an intelligent digital twin framework for microgrid security”. In: *2019 north American power symposium (NAPS)*. IEEE. 2019, pp. 1–6 (cit. on pp. 34, 35).

(Dascal et al., 2017) J. Dascal, M. Reid, W. W. IsHak, B. Spiegel, J. Recacho, B. Rosen and I. Danovitch. “Virtual reality and medical inpatients: a systematic review of randomized, controlled trials”. In: *Innovations in clinical neuroscience* 14.1-2 (2017), p. 14 (cit. on p. 42).

(David et al., 1994) R. David and H. Alla. “Petri nets for modeling of dynamic systems: A survey”. In: *Automatica* 30.2 (1994), pp. 175–202 (cit. on p. 83).

(De Bleser et al., 2006) L. De Bleser, R. Depreitere, K. D. WAELE, K. Vanhaecht, J. Vlayen and W. Sermeus. “Defining pathways”. In: *Journal of nursing management* 14.7 (2006), pp. 553–563 (cit. on pp. 9, 10, 12).

(DeRienzo et al., 2017) C. M. DeRienzo, R. J. Shaw, P. Meanor, E. Lada, J. Ferranti and D. Tanaka. “A discrete event simulation tool to support and predict hospital and clinic staffing”. In: *Health informatics journal* 23.2 (2017), pp. 124–133 (cit. on pp. 15, 17).

(Despiau et al., 2015) F. Despiau, M. Bombail, G. Le Duff, N. Peoc’h, R. Labatut and C. Ceaux. “The patient pathway, an essential element of the care project”. In: *Soins; la revue de reference infirmiere* 795 (2015), pp. 28–30 (cit. on p. 8).

(Dey et al., 2013) B. Dey, A. Mitra, K. Prakash, A. Basu, S. Ray and A. Mitra. “Effectiveness of care pathways in ideal hospital management system: a review”. In: *Indo global journal of pharmaceutical dciences* 3.3 (2013), pp. 185–191 (cit. on pp. 8, 12).

(Dickinson et al., 2000) C. Dickinson and M. Noud. “The antenatal ward care delivery map: a team model approach”. In: *Australian health review* 23.3 (2000), pp. 68–77 (cit. on p. 8).

(dictionary, 2011) S. medical dictionary. *Clinical pathway | definition of clinical pathway by Medical dictionary*. https://medical-dictionary.thefreedictionary.com/clinical+pathway. (Accessed on 07/13/2021). 2011 (cit. on p. 10).

(Dictionary, 2011) S. M. Dictionary. *Multidisciplinary | definition of multidisciplinary by Medical dictionary*. https://medical-dictionary.thefreedictionary.com/multidisciplinary. (Accessed on 07/27/2021). 2011 (cit. on p. 11).

(Donaldson et al., 2000) M. S. Donaldson, J. M. Corrigan, L. T. Kohn et al. *To err is human: building a safer health system*. Vol. 6. National academies press, 2000 (cit. on p. 12).

(Duguay et al., 2007) C. Duguay and F. Chetouane. “Modeling and improving emergency department systems using discrete event simulation”. In: *Simulation* 83.4 (2007), pp. 311–320 (cit. on pp. 16, 17).

(El Saddik, 2018) A. El Saddik. “Digital twins: the convergence of multimedia technologies”. In: *IEEE multimedia* 25.2 (2018), pp. 87–92 (cit. on pp. 27, 31).

(El Saddik et al., 2019) A. El Saddik, H. Badawi, R. A. M. Velazquez, F. Laamarti, R. G. Diaz, N. Bagaria and J. S. Arteaga-Falconi. “Dtwins: a digital twins ecosystem for health and well-being”. In: *MMTC communications-frontiers* (2019) (cit. on pp. 23, 24).

(Farsi et al., 2020) M. Farsi, A. Daneshkhah, A. Hosseinian-Far, H. Jahankhani et al. *Digital twin technologies and smart cities*. Springer, 2020 (cit. on p. 24).

(Fei et al., 2018) T. Fei, C. Jiangfeng, Q. Qinglin, M. Zhang, H. Zhang and S. Fangyuan. “Digital twin-driven product design, manufacturing and service with big data”. In: *The international journal of advanced manufacturing technology* 94.9-12 (2018), pp. 3563–3576 (cit. on pp. 27, 31).

(FlexSim, 2021) FlexSim. *3D simulation modeling and analysis software | FlexSim*. https://www.flexsim.com/. (Accessed on 08/05/2021). 2021 (cit. on p. 54).

(Fluxicon, 2021) Fluxicon. *Process mining and automated process discovery software for professionals - Fluxicon Disco*. https://fluxicon.com/disco/. (Accessed on 08/05/2021). 2021 (cit. on pp. 54, 117).

(Fujimoto et al., 2002) R. Fujimoto, D. Lunceford, E. Page and A. M. Uhrmacher. “Grand challenges for modeling and simulation”. In: *Schloss dagstuhl* 350 (2002) (cit. on p. 26).

(Gabriel et al., 2019) G. T. Gabriel, A. T. Campos, A. de Lima Magacho, L. C. Segismondi, F. F. Vilela, J. A. de Queiroz and J. A. B. Montevechi. *Lean healthcare integrated with discrete event simulation and design of experiments: an emergency department expansion.* Tech. rep. PeerJ preprints, 2019 (cit. on p. 13).

(Gartner, 2019) Gartner. *Gartner survey reveals digital twins are entering mainstream use.* https://www.gartner.com/en/newsroom/press-releases/2019-02-20-gartner-survey-reveals-digital-twins-are-entering-mai. (Accessed on 07/12/2021). Feb. 2019 (cit. on p. 41).

(Gaushell et al., 1987) D. J. Gaushell and H. T. Darlington. "Supervisory control and data acquisition". In: *Proceedings of the IEEE* 75.12 (1987), pp. 1645–1658 (cit. on p. 2).

(Gehrmann et al., 2019) C. Gehrmann and M. Gunnarsson. "A digital twin based industrial automation and control system security architecture". In: *IEEE transactions on industrial informatics* 16.1 (2019), pp. 669–680 (cit. on pp. 34, 35).

(Glaessgen et al., 2012) E. Glaessgen and D. Stargel. "The digital twin paradigm for future NASA and US Air Force vehicles". In: *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*. 2012, p. 1818 (cit. on p. 20).

(Gockel et al., 2012) B. Gockel, A. Tudor, M. Brandyberry, R. Penmetsa and E. Tuegel. "Challenges with structural life forecasting using realistic mission profiles". In: *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*. 2012, p. 1813 (cit. on p. 21).

(Graessler et al., 2017) I. Graessler and A. Pöhler. "Integration of a digital twin as human representation in a scheduling procedure of a cyber-physical production system". In: *2017 IEEE international conference on industrial engineering and engineering management (IEEM)*. IEEE. 2017, pp. 289–293 (cit. on p. 20).

(Grieves, 2003) M. Grieves. "Digital twin: manufacturing excellence through virtual factory replication". In: (2003) (cit. on p. 20).

(Grieves, 2014) M. Grieves. "Digital twin: manufacturing excellence through virtual factory replication". In: *White paper* 1 (2014), pp. 1–7 (cit. on p. 34).

(Grieves et al., 2017) M. Grieves and J. Vickers. "Digital twin: mitigating unpredictable, undesirable emergent behavior in complex systems". In: *Transdisciplinary perspectives on complex systems*. Springer, 2017, pp. 85–113 (cit. on p. 20).

(Grolinger et al., 2016) K. Grolinger, A. L'Heureux, M. A. Capretz and L. Seewald. "Energy forecasting for event venues: big data and prediction accuracy". In: *Energy and buildings* 112 (2016), pp. 222–233 (cit. on p. 70).

(Gualandi et al., 2019) R. Gualandi, C. Masella, D. Viglione and D. Tartaglini. "Exploring the hospital patient journey: What does the patient experience?" In: *PloS one* 14.12 (2019), e0224899 (cit. on p. 8).

(Guus Schrijvers, 2012) N. H. Guus Schrijvers Arjan van Hoorn. "The care pathway concept: concepts and theories: an introduction". In: 12 (2012) (cit. on p. 12).

(Hägglund et al., 2005) M. Hägglund, I. Scandurra and S. Koch. "Integrated care plan for shared home care: the technical architecture". In: *J inform technol healthcare* 3 (2005), 287e94 (cit. on p. 8).

(Haines, 2006) W. Haines. "Consequentialism". In: (2006) (cit. on p. 141).

(Halawa et al., 2020) F. Halawa, H. Daoud, I. G. Lee, Y. Li, S. W. Yoon and S. H. Chung. "Introduction of a real time location system to enhance the warehouse safety and operational efficiency". In: *International journal of production economics* 224 (2020), p. 107541 (cit. on p. 41).

(Halvorsrud et al., 2019) R. Halvorsrud, A. L. Lillegaard, M. Røhne and A. M. Jensen. "Managing complex patient journeys in healthcare". In: *Service design and service thinking in healthcare and hospital management*. Springer, 2019, pp. 329–346 (cit. on pp. 8, 10, 13).

(Hammad et al., 2011) A. Hammad and C. Zhang. "Towards real-time simulation of construction activities considering spatio-temporal resolution requirements for improving safety and productivity". In: *Proceedings of the 2011 winter simulation conference (WSC)*. IEEE. 2011, pp. 3533–3544 (cit. on pp. 4, 151).

(Hanisch et al., 2005) A. Hanisch, J. Tolujew and T. Schulze. "Initialization of online simulation models". In: *Proceedings of the winter simulation conference, 2005*. IEEE. 2005, 9–pp (cit. on pp. 4, 18, 151).

(Harrell et al., 2000) C. Harrell, R. Bowden and B. K. Ghosh. *Simulation using promodel*. McGraw-Hill higher education, 2000 (cit. on p. 14).

(Higgins et al., 1997) G. A. Higgins, G. L. Merrill, L. J. Hettinger, C. R. Kaufmann, H. R. Champion and R. M. Satava. "New simulation technologies for surgical training and certification: current status and future projections". In: *Presence: teleoperators & virtual environments* 6.2 (1997), pp. 160–172 (cit. on p. 41).

(Hippold, 2019) S. Hippold. *How digital twins simplify the IoT - smarter with Gartner*. https://www.gartner.com/smarterwithgartner/how-digital-twins-simplify-the-iot/. (Accessed on 07/12/2021). Jan. 2019 (cit. on p. 41).

(Hochhalter et al., 2014) J. Hochhalter, W. P. Leser, J. A. Newman, V. K. Gupta, V. Yamakov, S. R. Cornell, S. A. Willard and G. Heber. "Coupling damage-sensing particles to the digitial twin concept". In: (2014) (cit. on p. 21).

(Hrúz et al., 2007) B. Hrúz and M. Zhou. "Advanced textbooks in control and signal processing: modeling and control of discrete-event dynamic systems". In: *ISBN-13: 9781846288722, Springer-Verlag London Limited* (2007) (cit. on pp. 83, 84).

(Huang et al., 2016) Z. Huang, W. Dong, L. Ji and H. Duan. "Predictive monitoring of clinical pathways". In: *Expert systems with applications* 56 (2016), pp. 227–241 (cit. on pp. 2, 147).

(Hudson, 2011) B. Hudson. "Ten years of jointly commissioning health and social care in England". In: *International journal of integrated care* 11.Special 10th anniversary edition (2011) (cit. on p. 10).

(INFORMS, 2017) INFORMS. *Johns hopkins hospital opens capacity command center*. http://analytics-magazine.org/johns-hopkins-hospital-opens-capacity-command-center/. (Accessed on 06/21/2020). Jan. 2017 (cit. on pp. 23, 24).

(Inieke, 2020) O. Inieke. "Internet of things in health care: the social, ethical, legal and professional implications". In: (2020) (cit. on p. 141).

(Ireson, 1997) C. L. Ireson. "Critical pathways: effectiveness in achieving patient outcomes". In: *JONA: the journal of nursing administration* 27.6 (1997), pp. 16–23 (cit. on p. 10).

(Jacobson et al., 2006) S. H. Jacobson, S. N. Hall and J. R. Swisher. "Discrete-event simulation of health care systems". In: *Patient flow: reducing delay in healthcare delivery*. Springer, 2006, pp. 211–252 (cit. on pp. 4, 15, 150).

(Jacoby et al., 2020) M. Jacoby and T. Usländer. "Digital twin and internet of things—current standards landscape". In: *Applied sciences* 10.18 (2020), p. 6519 (cit. on p. 41).

(JIS, 1982) JIS. *Graphical symbols for process chart*. Japanese standards association. July 1982 (cit. on p. 59).

(Johnsson, 1999) C. Johnsson. "A graphical language for batch control". PhD thesis. Lund university, 1999 (cit. on p. 83).

(Jones et al., 2020) D. Jones, C. Snider, A. Nassehi, J. Yon and B. Hicks. "Characterising the digital twin: a systematic literature review". In: *CIRP Journal of manufacturing science and technology* 29 (2020), pp. 36–52 (cit. on pp. 27, 31).

(Kammoun et al., 2014) A. Kammoun, T. Loukil and W. Hachicha. “The use of discrete event simulation in hospital supply chain management”. In: *2014 international conference on advanced logistics and transport (ICALT)*. IEEE. 2014, pp. 143–148 (cit. on pp. 4, 150).

(Karakra et al., 2020) A. Karakra, E. Lamine, F. Fontanili and J. Lamothe. “HospiT’Win: a digital twin framework for patients’ pathways real-time monitoring and hospital organizational resilience capacity enhancement”. In: *International workshop on innovative simulation for healthcare*. CAL-TEK Srl. 2020, pp. 62–71 (cit. on pp. 66, 80, 92, 93).

(Karakra et al., 2019) A. Karakra, F. Fontanili, E. Lamine and J. Lamothe. “HospiT’Win: a predictive simulation-based digital twin for patients pathways in hospital”. In: *2019 IEEE EMBS international conference on biomedical & health informatics (BHI)*. IEEE. 2019, pp. 1–4 (cit. on p. 49).

(Karakra et al., 2018) A. Karakra, F. Fontanili, E. Lamine, J. Lamothe and A. Taweel. “Pervasive computing integrated discrete event simulation for a hospital digital twin”. In: *2018 IEEE/ACS 15th international conference on computer systems and Applications (AICCSA)*. IEEE. 2018, pp. 1–6 (cit. on p. 12).

(Katz et al., 1993) D. Katz and S. Manivannan. “Exception management on a shop floor using online simulation”. In: *Proceedings of the 25th conference on Winter simulation*. 1993, pp. 888–896 (cit. on p. 46).

(Kinsman et al., 2010) L. Kinsman, T. Rotter, E. James, P. Snow and J. Willis. “What is a clinical pathway? development of a definition to inform the debate”. In: *BMC medicine* 8.1 (2010), pp. 1–3 (cit. on pp. 8, 12).

(Kitchiner et al., 1996) D. Kitchiner, C. Davidson and P. Bundred. “Integrated care pathways: effective tools for continuous evaluation of clinical practice”. In: *Journal of evaluation in clinical practice* 2.1 (1996), pp. 65–69 (cit. on p. 12).

(Kohl et al., 2009) P. Kohl and D. Noble. “Systems biology and the virtual physiological human”. In: *Molecular systems biology* 5.1 (2009) (cit. on pp. 22, 24).

(Kondylakis et al., 2015) H. Kondylakis, E. G. Spanakis, S. Sfakianakis, V. Sakkalis, M. Tsiknakis, K. Marias, X. Zhao, H. Q. Yu and F. Dong. “Digital patient: personalized and translational data management through the MyHealthAvatar EU project”. In: *2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE. 2015, pp. 1397–1400 (cit. on pp. 23, 24).

(Kramer et al., 2020) A. Kramer, C. Dosi, M. Iori and M. Vignoli. “Successful implementation of discrete event simulation: the case of an Italian emergency department”. In: *arXiv preprint arXiv:2006.13062* (2020) (cit. on pp. 16, 17).

(Larsen, 2012) T. Larsen. “Guest editorial note on integrated care pathways: a basic tool for triple aim improvement of healthcare by collaborative local action”. In: *International journal of integrated care* 12.Special edition integrated care pathways (2012) (cit. on p. 10).

(Leape et al., 2005) L. L. Leape and D. M. Berwick. “Five years after to err is human: what have we learned?” In: *Jama* 293.19 (2005), pp. 2384–2390 (cit. on p. 12).

(Lee et al., 2013a) J. Lee, E. Lapira, B. Bagheri and H.-a. Kao. “Recent advances and trends in predictive manufacturing systems in big data environment”. In: *Manufacturing letters* 1.1 (2013), pp. 38–41 (cit. on p. 21).

(Lee et al., 2013b) J. Lee, E. Lapira, S. Yang and A. Kao. “Predictive manufacturing system-Trends of next-generation production systems”. In: *Ifac proceedings volumes* 46.7 (2013), pp. 150–156 (cit. on p. 20).

(Leng et al., 2019) J. Leng, H. Zhang, D. Yan, Q. Liu, X. Chen and D. Zhang. “Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop”. In: *Journal of ambient intelligence and humanized computing* 10.3 (2019), pp. 1155–1166 (cit. on pp. 24, 34, 35).

(Lin et al., 2019) W. Lin and M. Y. Low. “Concept and implementation of a cyber-pbysical digital twin for a SMT line”. In: *2019 IEEE international conference on industrial engineering and engineering management (IEEM)*. IEEE. 2019, pp. 1455–1459 (cit. on p. 34).

(S. Liu et al., 2020) S. Liu, Y. Li, K. P. Triantis, H. Xue and Y. Wang. "The diffusion of discrete event simulation approaches in health care management in the past four decades: a comprehensive review". In: *MDM policy & practice* 5.1 (2020), p. 2381468320915242 (cit. on p. 14).

(Y. Liu et al., 2019) Y. Liu, L. Zhang, Y. Yang, L. Zhou, L. Ren, F. Wang, R. Liu, Z. Pang and M. J. Deen. "A novel cloud-based framework for the elderly healthcare services using digital twin". In: *IEEE access* 7 (2019), pp. 49088–49101 (cit. on pp. 23, 24).

(Low et al., 2005) M. Y. H. Low, K. W. Lye, P. Lendermann, S. J. Turner, R. T. W. Chim and S. H. Leo. "An agent-based approach for managing symbiotic simulation of semiconductor assembly and test operation". In: *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. 2005, pp. 85–92 (cit. on pp. 44, 46).

(Ma et al., 2021) X. Ma, J. Cheng, Q. Qi and F. Tao. "Artificial intelligence enhanced interaction in digital twin shop-floor". In: *Procedia CIRP* 100 (2021), pp. 858–863 (cit. on p. 42).

(Madni et al., 2019) A. M. Madni, C. C. Madni and S. D. Lucero. "Leveraging digital twin technology in model-based systems Engineering". en. In: *Systems* 7.1 (Mar. 2019), p. 7 (cit. on pp. 2, 22, 149).

(Mahulea et al., 2014) C. Mahulea, L. Mahulea, J.-M. Garcıá-Soriano and J.-M. Colom. "Petri nets with resources for modeling primary healthcare systems". In: *2014 18th international conference on system theory, control and computing (ICSTCC)*. IEEE. 2014, pp. 639–644 (cit. on p. 13).

(Manning et al., 2017) V. Manning, J. B. Garfield, D. Best, L. Berends, R. Room, J. Mugavin, A. Larner, T. Lam, P. Buykx, S. Allsop et al. "Substance use outcomes following treatment: findings from the Australian Patient Pathways Study". In: *Australian & New Zealand journal of psychiatry* 51.2 (2017), pp. 177–189 (cit. on p. 8).

(Manyika et al., 2015) J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin and D. Aharon. "Unlocking the potential of the internet of things". In: *McKinsey global institute* 1 (2015) (cit. on p. 40).

(Martinez-Velazquez et al., 2019) R. Martinez-Velazquez, R. Gamez and A. El Saddik. "Cardio twin: a digital twin of the human heart running on the edge". In: *2019 IEEE international symposium on medical measurements and applications (MeMeA)*. IEEE. 2019, pp. 1–6 (cit. on pp. 22, 24).

(Merriam-Webster, 2021) Merriam-Webster. *Parameter | definition of parameter by Merriam-Webster*. https://www.merriam-webster.com/dictionary/parameter. (Accessed on 08/06/2021). 2021 (cit. on p. 31).

(Miyamoto et al., 2005) T. Miyamoto and S. Kumagai. "A survey of object-oriented Petri nets and analysis methods". In: *IEICE transactions on fundamentals of electronics, communications and computer sciences* 88.11 (2005), pp. 2964–2971 (cit. on p. 84).

(Model, 2011) B. P. Model. "Notation (BPMN) version 2.0". In: *OMG specification, object management group* (2011), pp. 22–31 (cit. on p. 13).

(Montevechi et al., 2007) J. A. B. Montevechi, A. F. de Pinho, F. Leal and F. A. S. Marins. "Application of design of experiments on the simulation of a process in automotive industry". In: *2007 winter simulation conference*. IEEE. 2007, pp. 1601–1609 (cit. on p. 14).

(Moon, 2007) Y. B. Moon. "Enterprise resource planning (ERP): a review of the literature". In: *International journal of management and enterprise development* 4.3 (2007), pp. 235–264 (cit. on p. 2).

(Negri et al., 2019) E. Negri, L. Fumagalli, C. Cimino and M. Macchi. "FMU-supported simulation for CPS digital twin". In: *Procedia manufacturing* 28 (2019), pp. 201–206 (cit. on p. 41).

(Negri et al., 2017) E. Negri, L. Fumagalli and M. Macchi. “A Review of the roles of digital twin in CPS-based production systems”. In: *Procedia manufacturing* 11 (2017), pp. 939–948 (cit. on p. 20).

(Nüttgens et al., 1992) M. Nüttgens and A.-W. Scheer. “Semantische prozeßmodellierung auf der grundlage ereignisgesteuerter prozeßketten (EPK)”. In: *Veröffentlichungen des Instituts für wirtschaftsinformatik* 89 (1992) (cit. on p. 13).

(O’Hanlon, 2005) T. O’Hanlon. “Computerized maintenance management and enterprise asset management best practices”. In: *Reliabilityweb. com asset management white paper series, NetexpessUSA Inc* (2005) (cit. on p. 2).

(Panella et al., 2003) M. Panella, S. Marchisio and F. Di Stanislao. “Reducing clinical variations with clinical pathways: do pathways work?” In: *International journal for quality in health care* 15.6 (2003), pp. 509–521 (cit. on p. 12).

(Percival et al., 2006) J. Percival and J. Hanson. “Big brother or brave new world? telecare and its implications for older people’s independence and social inclusion”. In: *Critical Social Policy* 26.4 (2006), pp. 888–909 (cit. on p. 140).

(Petri, 1962) C. A. Petri. “Kommunikation mit automaten”. In: (1962) (cit. on p. 83).

(Pinder et al., 2005) R. Pinder, R. Petchey, S. Shaw and Y. Carter. “What’s in a care pathway? towards a cultural cartography of the new NHS”. In: *Sociology of health & illness* 27.6 (2005), pp. 759–779 (cit. on p. 8).

(Pitt et al., 2008) M. Pitt, D. Bensley, S. Brailsford, S. Burnell, T. Chaussalet, R. Davies, S. Dodds, A. Pollard, B. Wherry and D. Worthington. “Simulation for strategic planning in healthcare: the state of the art”. In: *Briefing paper produced for NHS institute* (2008) (cit. on p. 17).

(Ponomarev et al., 2017) K. Ponomarev, N. Kudryashov, N. Popelnukha and V. Potekhin. “Main principals and issues of digital twin development for complex technological processes”. In: *Proceedings of the 28th DAAAM international symposium*. 2017, pp. 0523–0528 (cit. on p. 34).

(Porritt et al., 2012) J. Porritt, S. Baker and Z. Marshman. “A service evaluation of patient pathways and care experiences of dentally anxious adult patients.” In: *Community dental health* 29.3 (2012), pp. 198–202 (cit. on p. 8).

(Preeti Wadhwani, 2021) S. L. Preeti Wadhwani. *Digital twin market size, growth forecast report 2027*. https://www.gminsights.com/industry-analysis/digital-twin-market. (Accessed on 07/12/2021). July 2021 (cit. on p. 40).

(Qi et al., 2018) Q. Qi, F. Tao, Y. Zuo and D. Zhao. “Digital twin service towards smart manufacturing”. In: *Procedia cirp* 72 (2018), pp. 237–242 (cit. on p. 34).

(Rathore et al., 2021) M. M. Rathore, S. A. Shah, D. Shukla, E. Bentafat and S. Bakiras. “The role of AI, machine learning, and big data in digital twinning: A systematic literature review, challenges, and opportunities”. In: *IEEE Access* 9 (2021), pp. 32030–32052 (cit. on p. 42).

(Raunak et al., 2009) M. Raunak, L. Osterweil, A. Wise, L. Clarke and P. Henneman. “Simulating patient flow through an emergency department using process-driven discrete event simulation”. In: *2009 ICSE workshop on software engineering in health care*. IEEE. 2009, pp. 73–83 (cit. on pp. 16, 17).

(Redelinghuys et al., 2018) A. Redelinghuys, A. Basson and K. Kruger. “A six-layer digital twin architecture for a manufacturing cell”. In: *International workshop on service orientation in holonic and multi-agent manufacturing*. Springer. 2018, pp. 412–423 (cit. on pp. 34, 35).

(Richter et al., 2019) P. Richter and H. Schlieter. “Understanding patient pathways in the context of integrated health care services-implications from a scoping review”. In: (2019) (cit. on p. 8).

(Rodríguez-Aguilar et al., 2020) R. Rodríguez-Aguilar and J.-A. Marmolejo-Saucedo. “Conceptual framework of digital health public emergency system: digital twins and multiparadigm simulation”. In: *EAI endorsed transactions on pervasive health and technology* 6.21 (2020), e3 (cit. on pp. 27, 31).

(Roland Rosen, 2018) A. S. Roland Rosen Stefan Boschert. “Next Generation Digital Twin”. In: *atp magazin* 60.10 (2018), pp. 86–96 (cit. on pp. 2, 149).

(Rosen et al., 2015) R. Rosen, G. Von Wichert, G. Lo and K. D. Bettenhausen. “About the importance of autonomy and digital twins for the future of manufacturing”. In: *IFAC-PapersOnLine* 48.3 (2015), pp. 567–572 (cit. on p. 21).

(Saenz de Ugarte et al., 2009) B. Saenz de Ugarte, A. Artiba and R. Pellerin. “Manufacturing execution system–a literature review”. In: *Production planning and control* 20.6 (2009), pp. 525–539 (cit. on p. 2).

(Salleh et al., 2017) S. Salleh, P. Thokala, A. Brennan, R. Hughes and A. Booth. “Simulation modelling in healthcare: an umbrella review of systematic literature reviews”. In: *PharmacoEconomics* 35.9 (2017), pp. 937–949 (cit. on p. 4).

(Saracco, 2019) R. Saracco. “Digital Twins: bridging physical space and cyberspace”. In: *Computer* 52.12 (2019), pp. 58–64 (cit. on pp. 2, 149).

(Sarno et al., 2016) D. Sarno and M. E. Nenni. “Daily nurse requirements planning based on simulation of patient flows”. In: *Flexible services and manufacturing journal* 28.3 (2016), pp. 526–549 (cit. on p. 8).

(Scharff, 2018) S. Scharff. *From digital twin to improved patient experience - siemens healthineers global*. https://www.siemens-healthineers.com/news/mso-digital-twin-mater.html. Sept. 2018 (cit. on pp. 23, 24).

(Schrijvers et al., 2012) G. Schrijvers, A. van Hoorn and N. Huiskes. “The care pathway: concepts and theories: an introduction”. In: *International journal of integrated care* 12.Special edition integrated care pathways (2012) (cit. on pp. 8, 10, 12).

(Sethi et al., 2017) P. Sethi and S. R. Sarangi. “Internet of things: architectures, protocols, and applications”. In: *Journal of electrical and computer engineering* 2017 (2017) (cit. on p. 40).

(Shafto et al., 2012) M. Shafto, M. Conroy, R. Doyle, E. Glaessgen, C. Kemp, J. LeMoigne and L. Wang. “Modeling, simulation, information technology & processing roadmap”. In: *National aeronautics and space administration* (2012) (cit. on p. 21).

(Shannon, 1975) R. E. Shannon. *Systems simulation; the art and science*. Tech. rep. 1975 (cit. on p. 41).

(Siemens, 2020) Siemens. *Simulation & Digital Twin*. https://assets.new.siemens.com/siemens/assets/api/uuid:ce77eef2-3ae4-4910-9d88-e57697aa0196/SimulationDigitalTwinOutlook2030.pdf. (Accessed on 08/09/2021). 2020 (cit. on p. 41).

(Siemens, 2021) Siemens. *The Birth of Constant Optimization*. https://new.siemens.com/global/en/company/stories/research-technologies/digitaltwin/digital-twin.html. (Accessed on 08/09/2021). 2021 (cit. on p. 41).

(Singh et al., 2021) M. Singh, E. Fuenmayor, E. P. Hinchy, Y. Qiao, N. Murray and D. Devine. “Digital twin: origin to future”. In: *Applied system innovation* 4.2 (2021), p. 36 (cit. on pp. 27, 31).

(Slaich et al., 2018) B. Slaich, A. Careless and M. Flynn. “University hospitals of leicester mesothelioma patient pathways April 2015 to March 2016”. In: *Lung cancer* 115 (2018), S28 (cit. on p. 8).

(Spyrou et al., 2005) S. Spyrou, P. Bamidis, K. Pappas and N. Maglaveras. “Extending UML activity diagrams for workflow modelling with clinical documents in regional health information systems”. In: *Connecting medical informatics and bioinformatics: proceedings of the 19th medical informatics Europe conference (MIE2005). Geneva, Switzerland*. 2005, pp. 1160–1165 (cit. on p. 13).

(Srivastava et al., 2020) S. R. Srivastava and S. Dubey. “Knowledge-based systems in medical applications”. In: *Computational intelligence and its applications in healthcare*. Elsevier, 2020, pp. 189–215 (cit. on p. 8).

(Stark, 2015) J. Stark. “Product lifecycle management”. In: *Product lifecycle management (Volume 1)*. Springer, 2015, pp. 1–29 (cit. on p. 2).

(Stefanini et al., 2016) A. Stefanini, D. Aloini, R. Dulmin and V. Mininno. “Linking diagnostic-related groups (DRGs) to their processes by process mining.” In: *HEALTHINF*. 2016, pp. 438–443 (cit. on pp. 8, 12).

(Steinmetz et al., 2018) C. Steinmetz, A. Rettberg, F. G. C. Ribeiro, G. Schroeder and C. E. Pereira. “Internet of things ontology for digital twin in cyber physical systems”. In: *2018 VIII Brazilian symposium on computing systems engineering (SBESC)*. IEEE. 2018, pp. 154–159 (cit. on p. 25).

(Sunyaev, 2020) A. Sunyaev. “Cloud computing”. In: *Internet computing*. Springer, 2020, pp. 195–236 (cit. on p. 42).

(Swancutt et al., 2017) D. Swancutt, S. Joel-Edgar, M. Allen, D. Thomas, H. Brant, J. Benger, R. Byng and J. Pinkney. “Not all waits are equal: an exploratory investigation of emergency care patient pathways”. In: *BMC health services research* 17.1 (2017), p. 436 (cit. on p. 8).

(Talkhestani et al., 2019) B. A. Talkhestani, T. Jung, B. Lindemann, N. Sahlab, N. Jazdi, W. Schloegl and M. Weyrich. “An architecture of an intelligent digital twin in a cyber-physical production system”. In: *at-Automatisierungstechnik* 67.9 (2019), pp. 762–782 (cit. on pp. 34, 35).

(Tao et al., 2019) F. Tao, F. Sui, A. Liu, Q. Qi, M. Zhang, B. Song, Z. Guo, S. C.-Y. Lu and A. Y. Nee. “Digital twin-driven product design framework”. In: *International journal of production research* 57.12 (2019), pp. 3935–3953 (cit. on pp. 24, 42).

(Tao et al., 2018a) F. Tao, H. Zhang, A. Liu and A. Y. Nee. “Digital twin in industry: state-of-the-art”. In: *IEEE transactions on industrial informatics* 15.4 (2018), pp. 2405–2415 (cit. on p. 34).

(Tao et al., 2018b) F. Tao, M. Zhang, Y. Liu and A. Y. Nee. “Digital twin driven prognostics and health management for complex equipment”. In: *Cirp annals* 67.1 (2018), pp. 169–172 (cit. on p. 34).

(Tavakoli et al., 2008) S. Tavakoli, A. Mousavi and A. Komashie. “A generic framework for real-time discrete event simulation (DES) modelling”. In: *2008 winter simulation conference*. IEEE. 2008, pp. 1931–1938 (cit. on pp. 4, 18, 19, 151).

(Thompson et al., 2015) C. A. Thompson, A. W. Kurian and H. S. Luft. “Linking electronic health records to better understand breast cancer patient pathways within and between two health systems”. In: *eGEMs* 3.1 (2015) (cit. on p. 8).

(Tlapa et al., 2020) D. Tlapa, C. A. Zepeda-Lugo, G. L. Tortorella, Y. A. Baez-Lopez, J. Limon-Romero, A. Alvarado-Iniesta and M. I. Rodriguez-Borbon. “Effects of lean healthcare on patient flow: a systematic review”. In: *Value in health* 23.2 (2020), pp. 260–273 (cit. on p. 8).

(Tritter et al., 2014) J. Q. Tritter, K. Lutfey and J. McKinlay. “What are tests for? The implications of stuttering steps along the US patient pathway”. In: *Social science & medicine* 107 (2014), pp. 37–43 (cit. on p. 8).

(Tukey, 1969) J. W. Tukey. “Analyzing data: sanctification or detective work?” In: *American psychologist* 24.2 (1969), p. 83 (cit. on p. 125).

(Van Der Aalst, 1998) W. M. Van Der Aalst. “Three good reasons for using a Petri-net-based workflow management system”. In: *Information and Process Integration in Enterprises*. Springer, 1998, pp. 161–182 (cit. on p. 84).

(VanDerHorn et al., 2021) E. VanDerHorn and S. Mahadevan. “Digital twin: generalization, characterization and implementation”. In: *Decision support systems* (2021), p. 113524 (cit. on pp. 28, 31).

(Vanhaecht et al., 2007) K. Vanhaecht, K. De Witte and W. Sermeus. “The impact of clinical pathways on the organisation of care processes. 2007”. In: *Leuven: ACCO* (2007) (cit. on p. 12).

(Vanhaecht, 2007) K. Vanhaecht. “The impact of clinical pathways on the organisation of care processes”. In: (2007) (cit. on p. 10).

(Vanhaecht et al., 2010) K. Vanhaecht, M. Panella, R. Van Zelm and W. Sermeus. “An overview on the history and concept of care pathways as complex interventions”. In: *International journal of care pathways* 14.3 (2010), pp. 117–123 (cit. on p. 8).

(Viceconti et al., 2008) M. Viceconti, G. Clapworthy and S. V. S. Jan. “The virtual physiological human—a european initiative for in silico human modelling—”. In: *The journal of physiological sciences* (2008), pp. 0810200082–0810200082 (cit. on pp. 22, 24).

(Vozenilek et al., 2004) J. Vozenilek, J. S. Huff, M. Reznek and J. A. Gordon. “See one, do one, teach one: advanced technology in medical education”. In: *Academic emergency medicine* 11.11 (2004), pp. 1149–1154 (cit. on p. 42).

(Wang, 2020) Z. Wang. “Digital twin technology”. In: *Industry 4.0-impact on intelligent logistics and manufacturing*. IntechOpen, 2020 (cit. on pp. 28, 31).

(Watson, 2018) M. Watson. “Engineering elegant systems: theory of systems engineering”. In: *NASA 4th draft* (2018) (cit. on p. 14).

(Wigfield et al., 1996) A. Wigfield and E. Boon. “Critical care pathway development: the way forward”. In: *British journal of nursing* 5.12 (1996), pp. 732–735 (cit. on p. 8).

(Wong et al., 2016) Z. S.-Y. Wong, A. C.-H. Lit, S.-Y. Leung, K.-L. Tsui and K.-S. Chin. “A discrete-event simulation study for emergency room capacity management in a Hong Kong hospital”. In: *2016 winter simulation conference (WSC)*. IEEE. 2016, pp. 1970–1981 (cit. on pp. 16, 17).

(World Bank, 2021) World Bank. *Current health expenditure (% of GDP) | data*. `https://data.worldbank.org/indicator/SH.XPD.CHEX.GD.ZS`. (Accessed on 07/13/2021). 2021 (cit. on pp. 1, 147).

(Wu et al., 2021) C. Wu, Y. Zhou, M. V. P. Pessôa, Q. Peng and R. Tan. “Conceptual digital twin modeling based on an integrated five-dimensional framework and TRIZ function model”. In: *Journal of manufacturing systems* 58 (2021), pp. 79–93 (cit. on p. 34).

(B. Zeigler et al., 2019) B. Zeigler, M. K. Traoré, G. Zacharewicz and R. Duboz. *Value-based learning healthcare systems: Integrative modeling and simulation*. Institution of Engineering and Technology, 2019 (cit. on pp. 10, 13).

(B. P. Zeigler et al., 2018) B. P. Zeigler, A. Muzy and E. Kofman. *Theory of modeling and simulation: discrete event & iterative system computational foundations*. Academic press, 2018 (cit. on p. 41).

(Zhang, 2018) X. Zhang. “Application of discrete event simulation in health care: a systematic review”. In: *BMC health services research* 18.1 (2018), pp. 1–11 (cit. on pp. 4, 150).

(Zheng et al., 2019) Y. Zheng, S. Yang and H. Cheng. “An application framework of digital twin and its case study”. In: *Journal of Ambient Intelligence and Humanized Computing* 10.3 (2019), pp. 1141–1153 (cit. on p. 34).

(Q. Zhu et al., 2018) Q. Zhu, S. Johnson and J. Sarkis. “Lean six sigma and environmental sustainability: a hospital perspective”. In: *Supply chain forum: an international journal*. Vol. 19. 1. Taylor & Francis. 2018, pp. 25–41 (cit. on p. 13).

(Z. Zhu et al., 2019) Z. Zhu, C. Liu and X. Xu. “Visualisation of the digital twin data in manufacturing by using augmented reality”. In: *Procedia cirp* 81 (2019), pp. 898–903 (cit. on pp. 27, 31).

(Z. Zhu et al., 2012) Z. Zhu, B. H. Hen and K. L. Teow. “Estimating ICU bed capacity using discrete event simulation”. In: *International journal of health care quality assurance* (2012) (cit. on pp. 4, 150).

(Zweifach et al., 2019) S. M. Zweifach and M. M. Triola. “Extended reality in medical education: driving adoption through provider-centered design”. In: *Digital biomarkers* 3.1 (2019), pp. 14–21 (cit. on p. 42).

Appendix: GRAFNet Charts

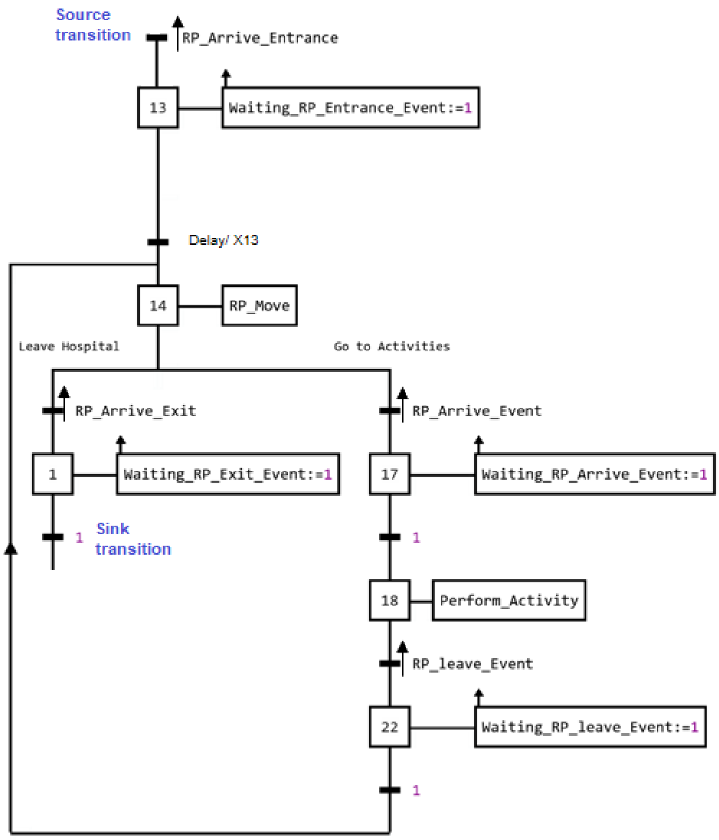


Figure 6.2: Generic GRAFNet for the real patient pathways

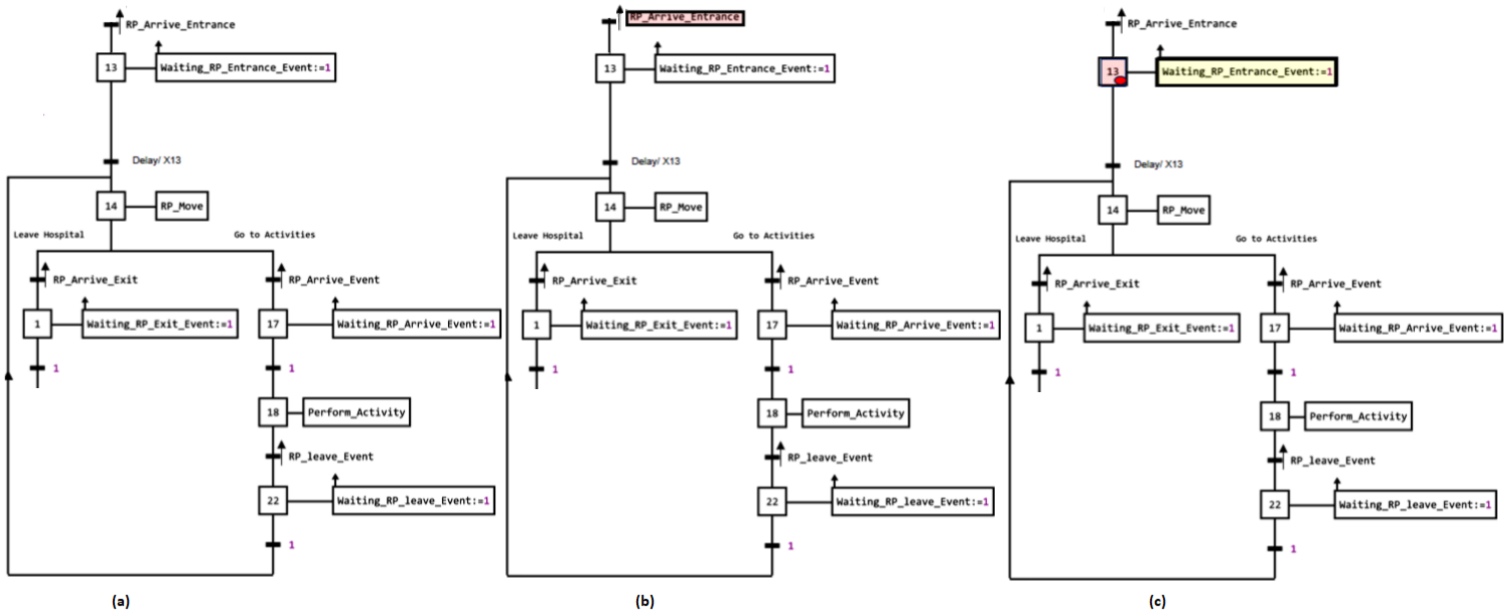


Figure 6.3: Sample execution for the generic GRAFNet real patient pathway: start executing the GRAFNet

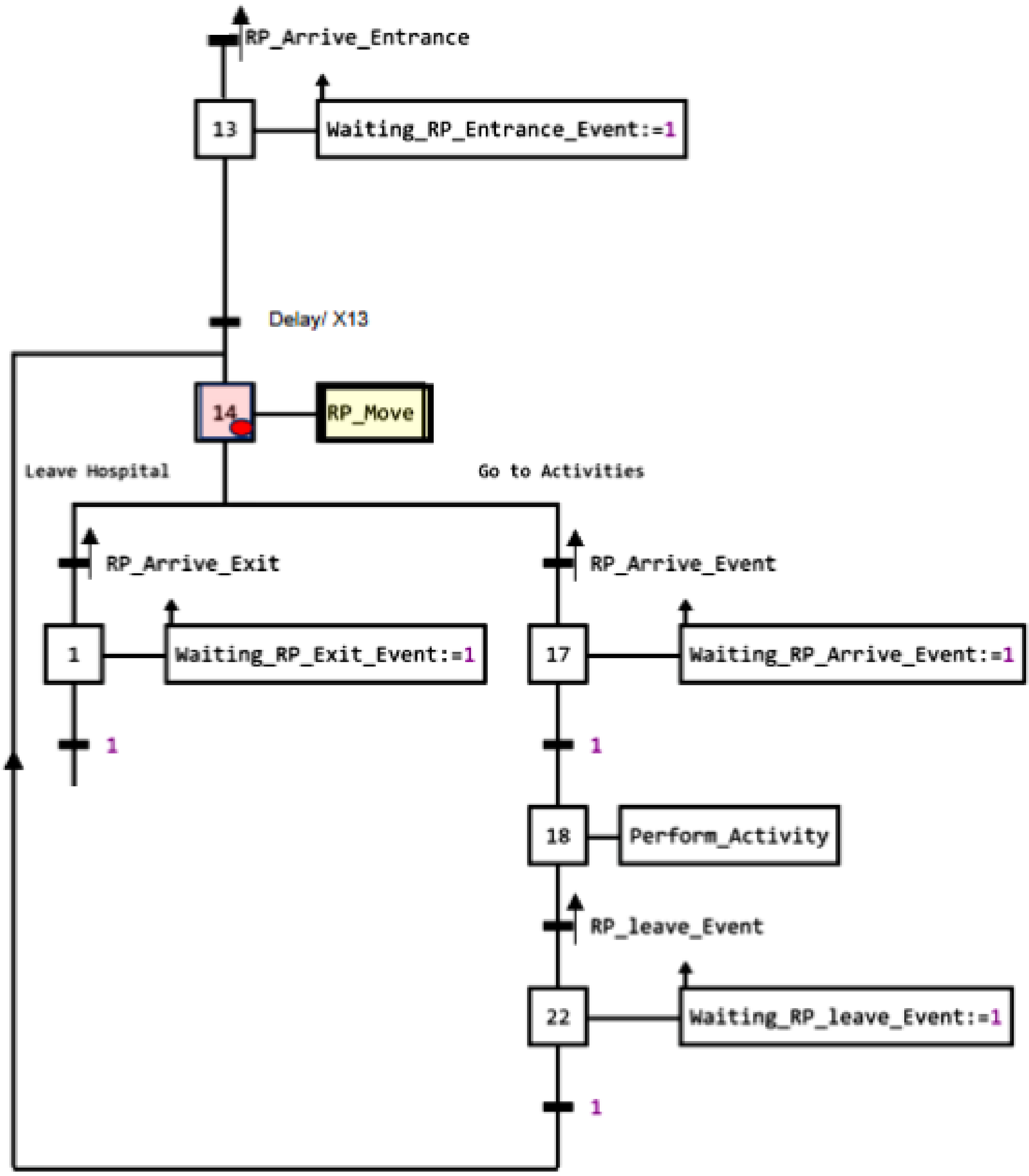


Figure 6.4: Sample execution for the generic GRAFNet real patient pathway: execute moving activity

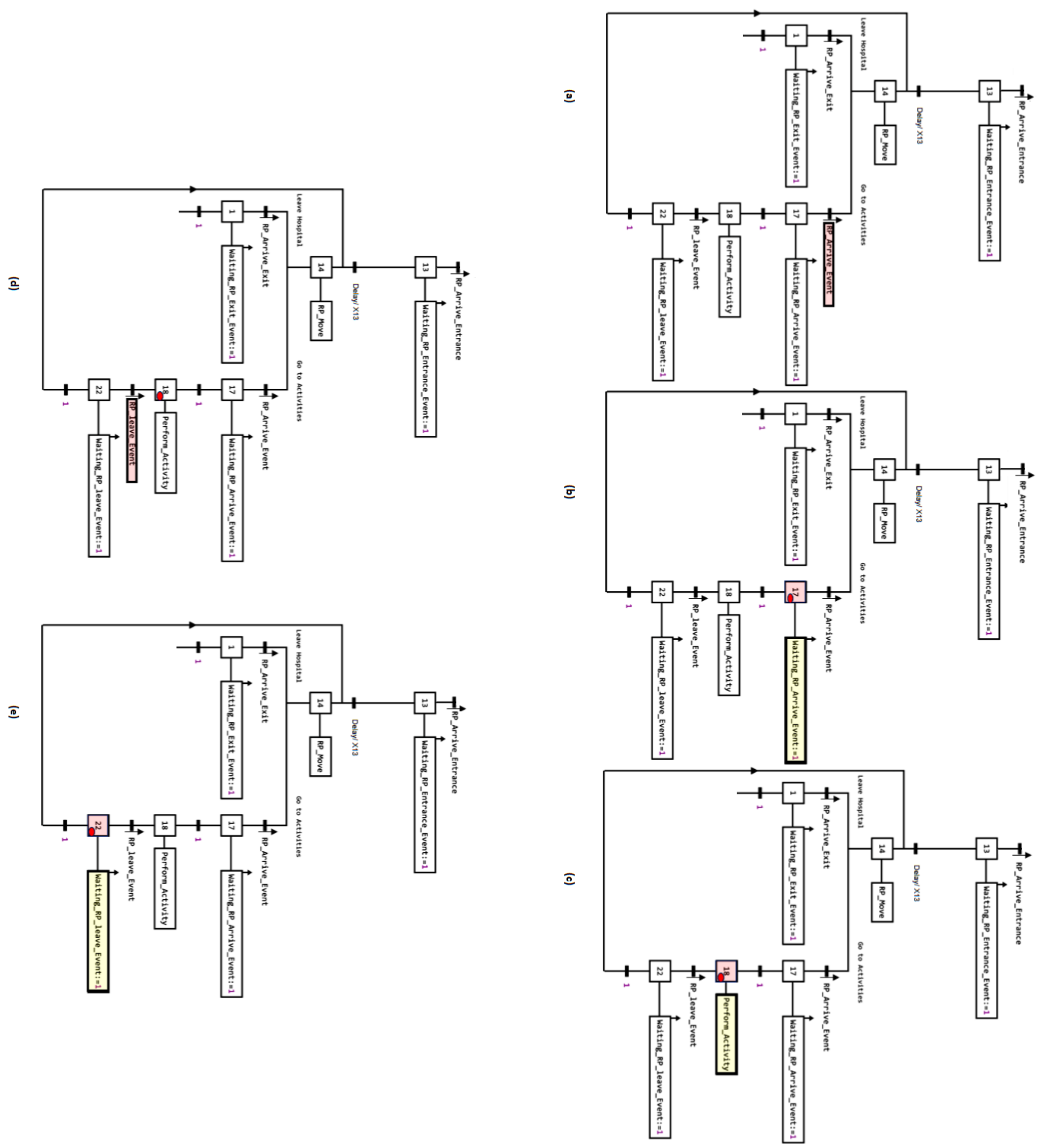


Figure 6.5: Sample execution for the generic GRAFNet real patient pathway: perform activity

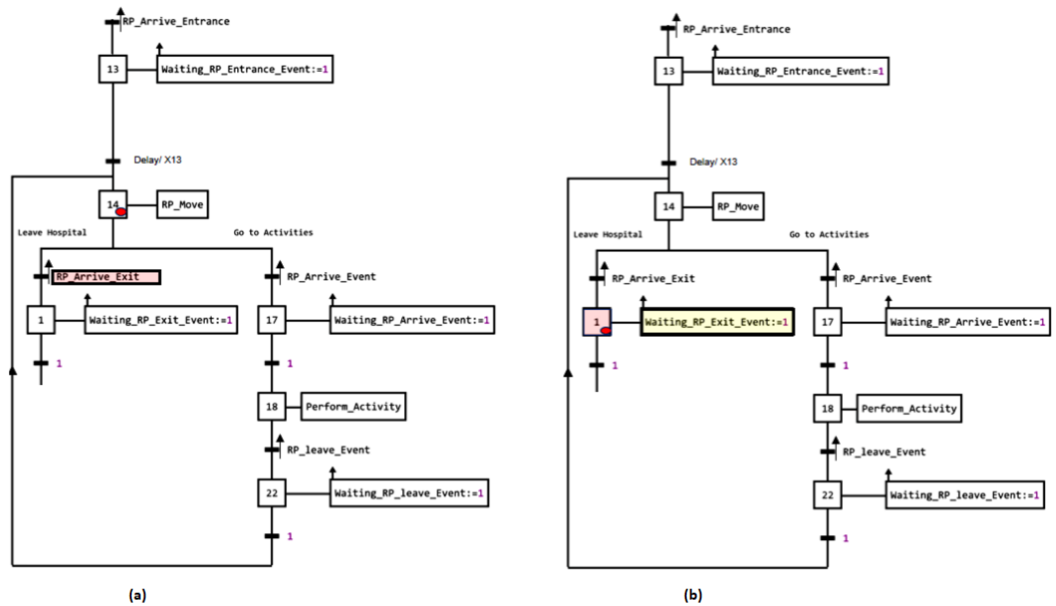


Figure 6.6: Sample execution for the generic GRAFNet real patient pathways: leave the hospital

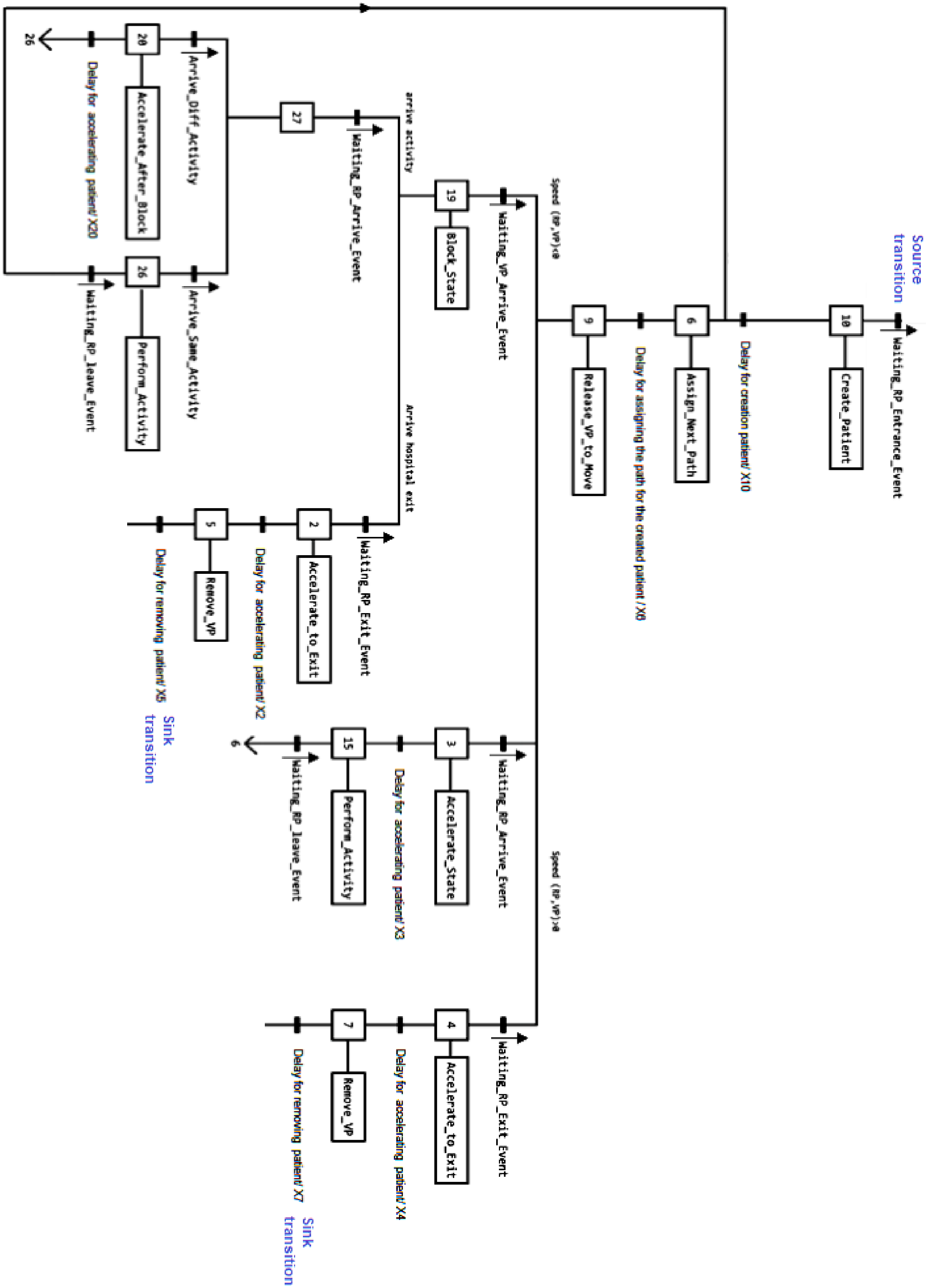


Figure 6.7: Generic GRAFNet for the virtual patient pathways

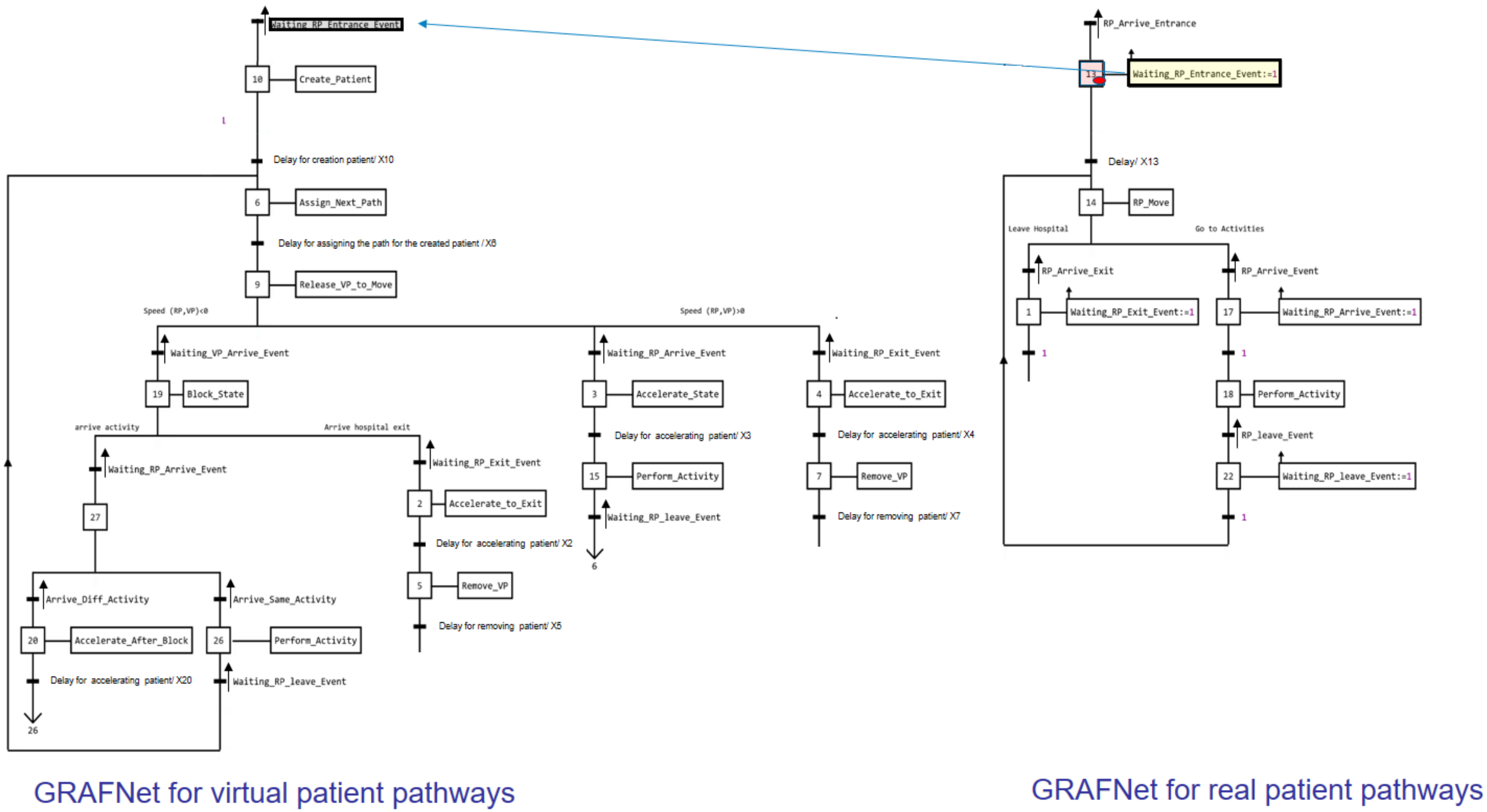


Figure 6.8: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 1

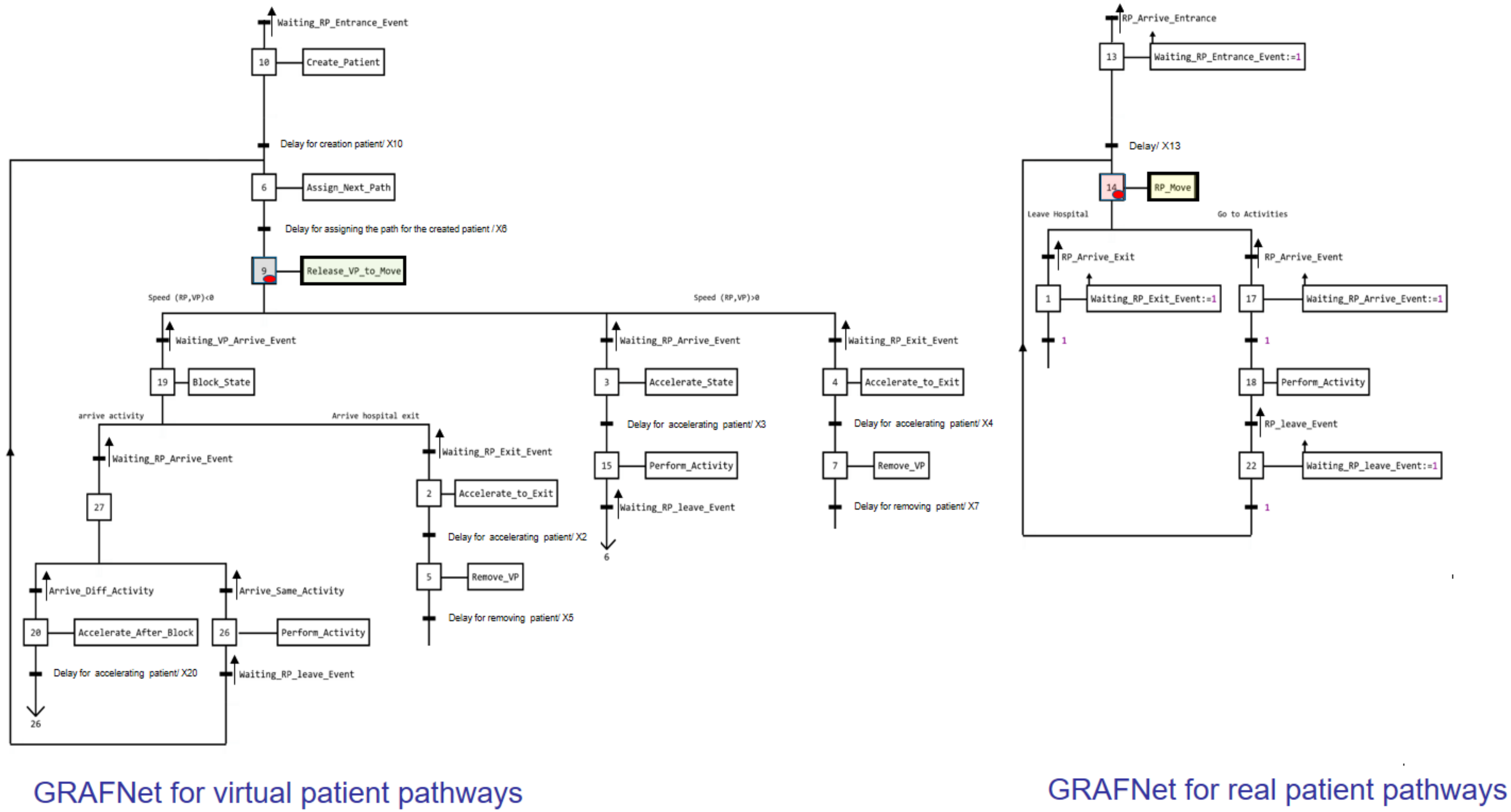


Figure 6.9: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 2

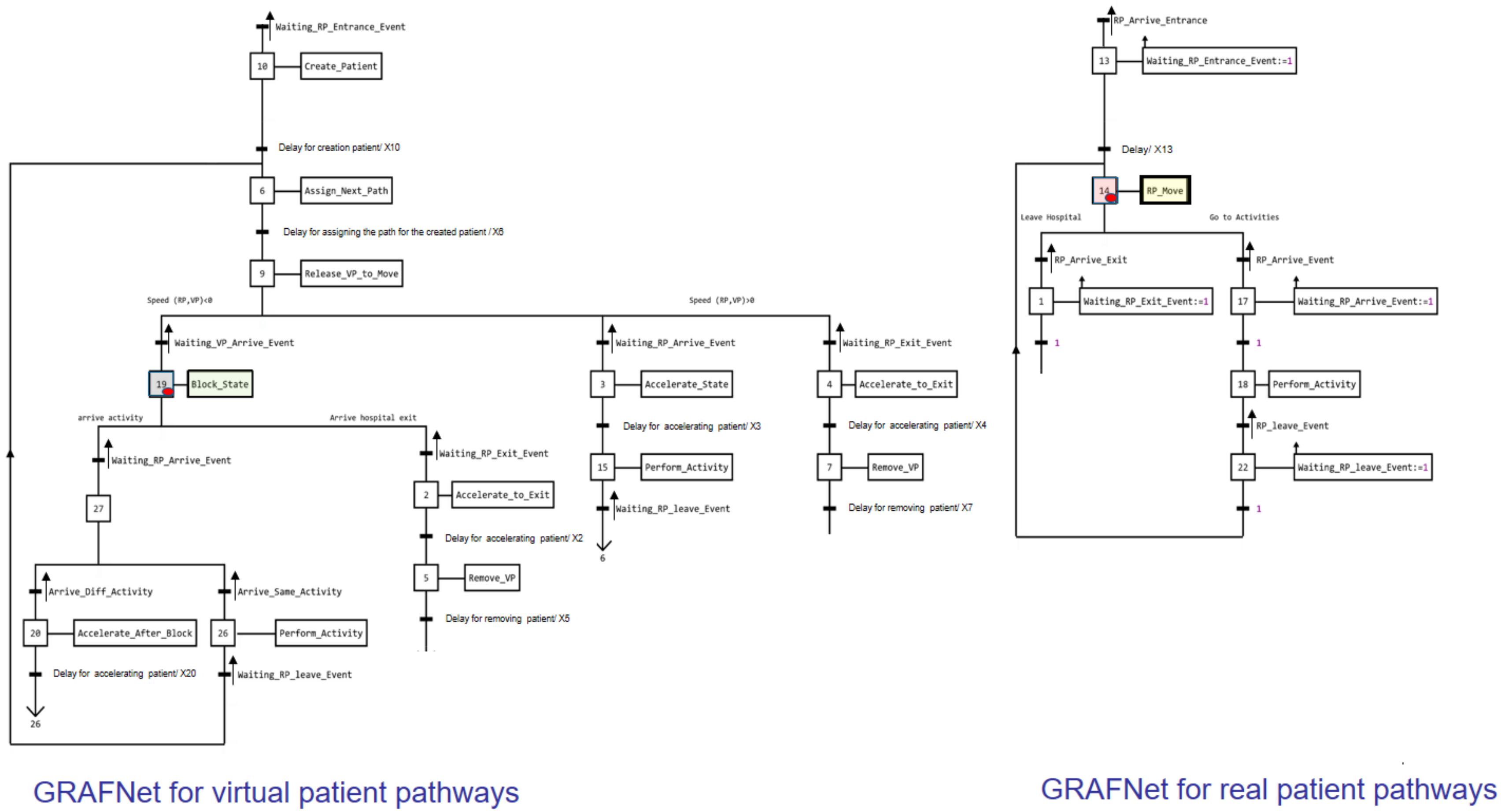


Figure 6.10: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 3

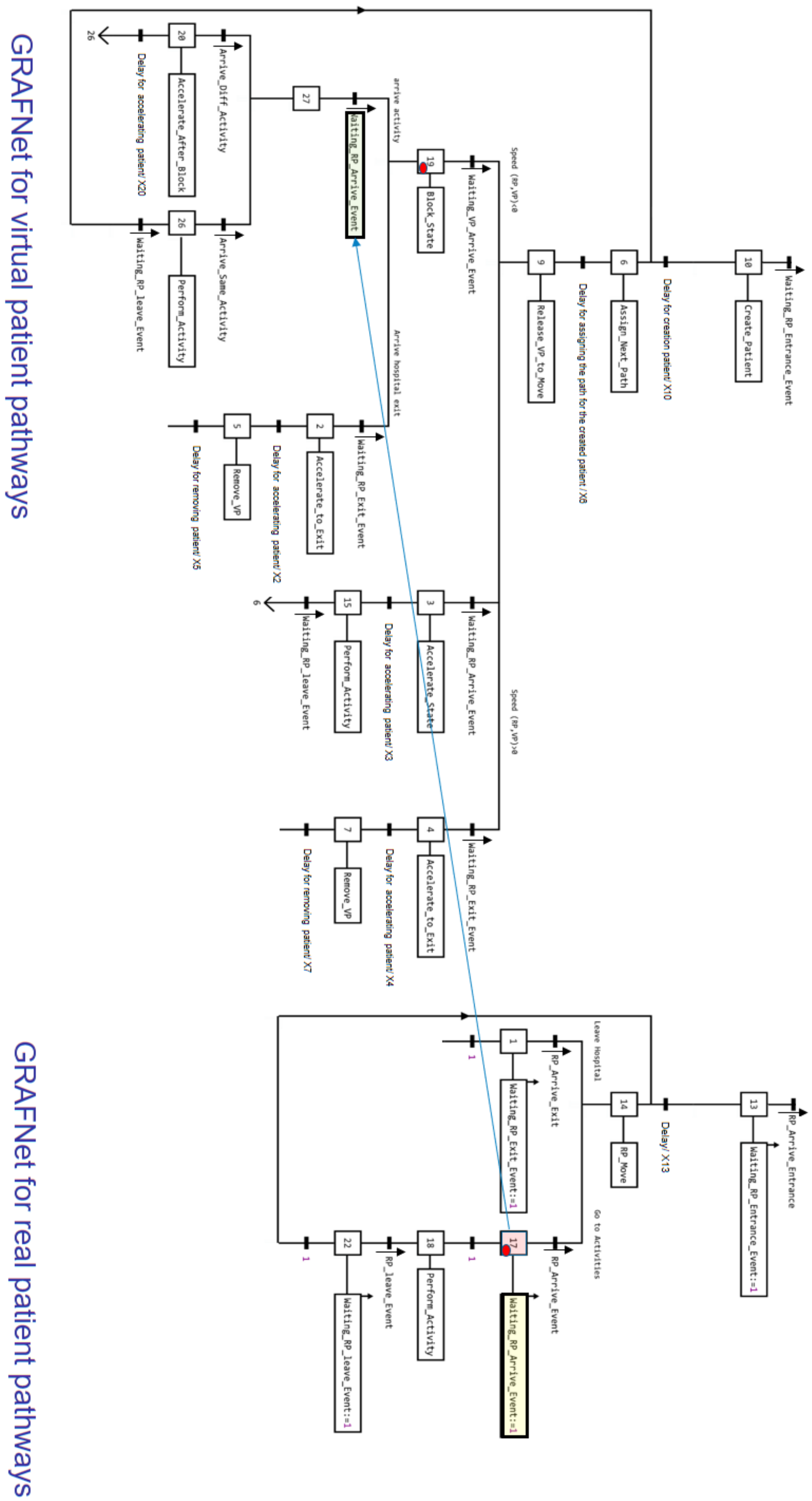


Figure 6.11: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 4

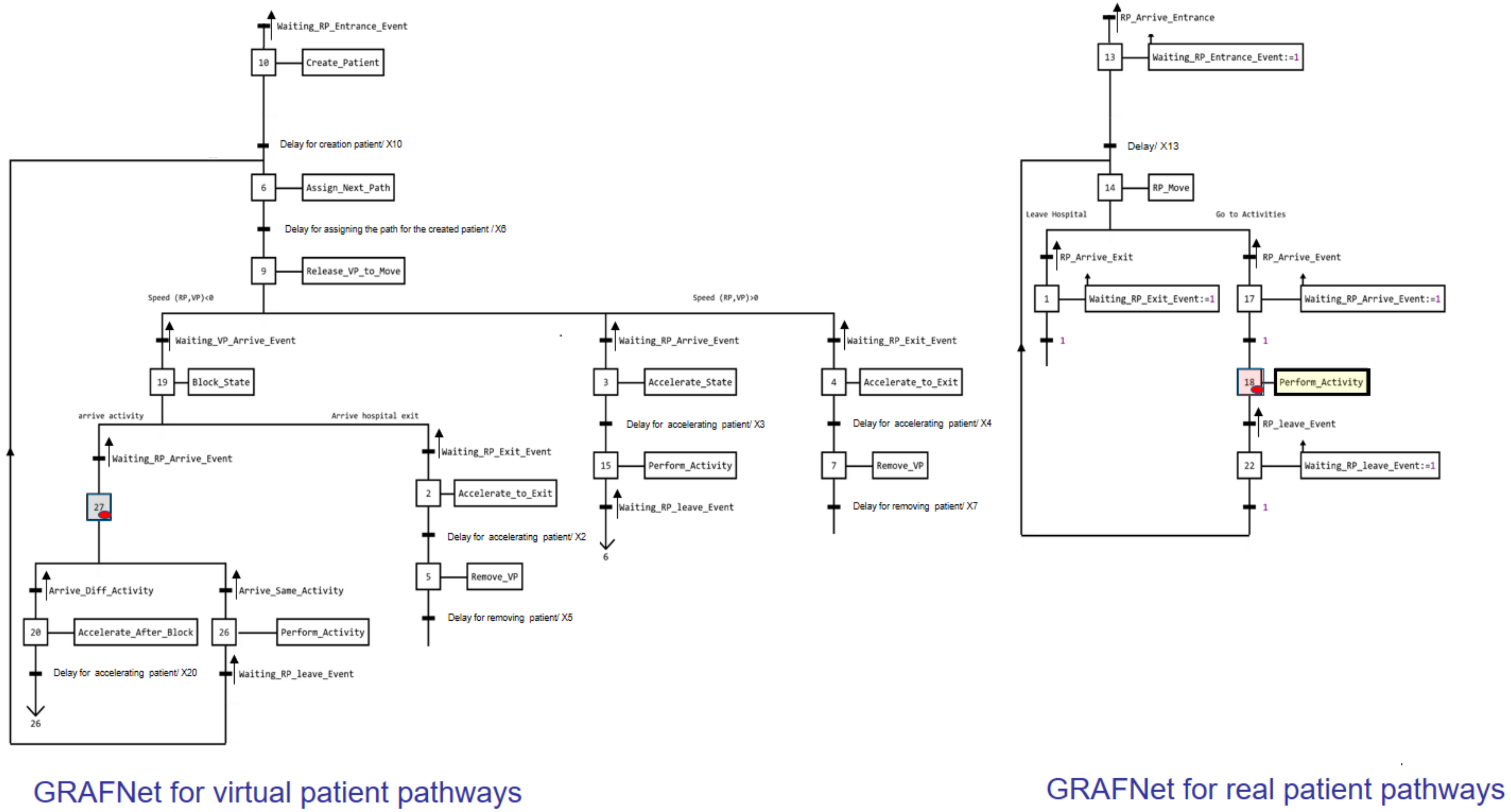


Figure 6.12: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 5

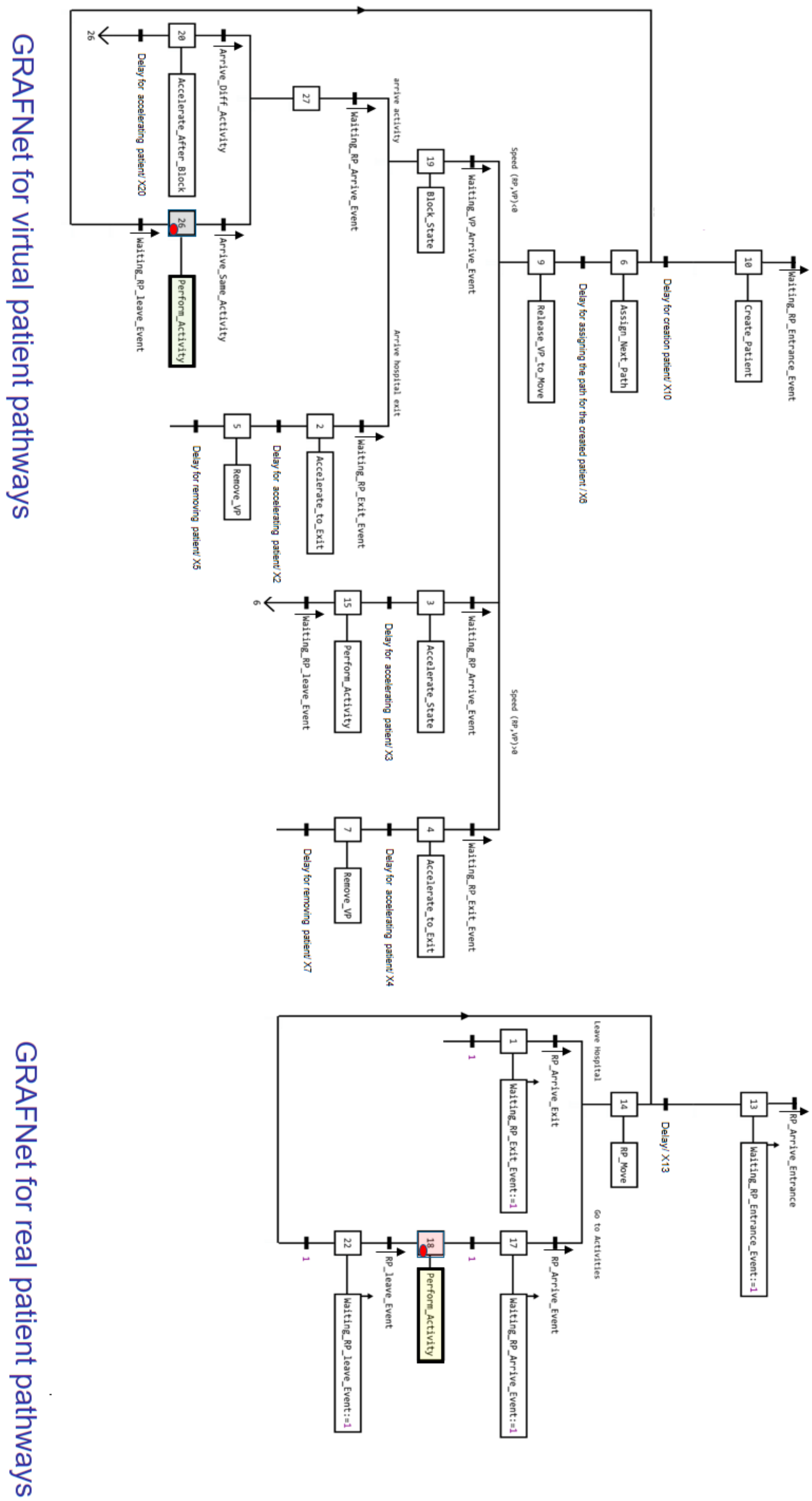


Figure 6.13: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 6

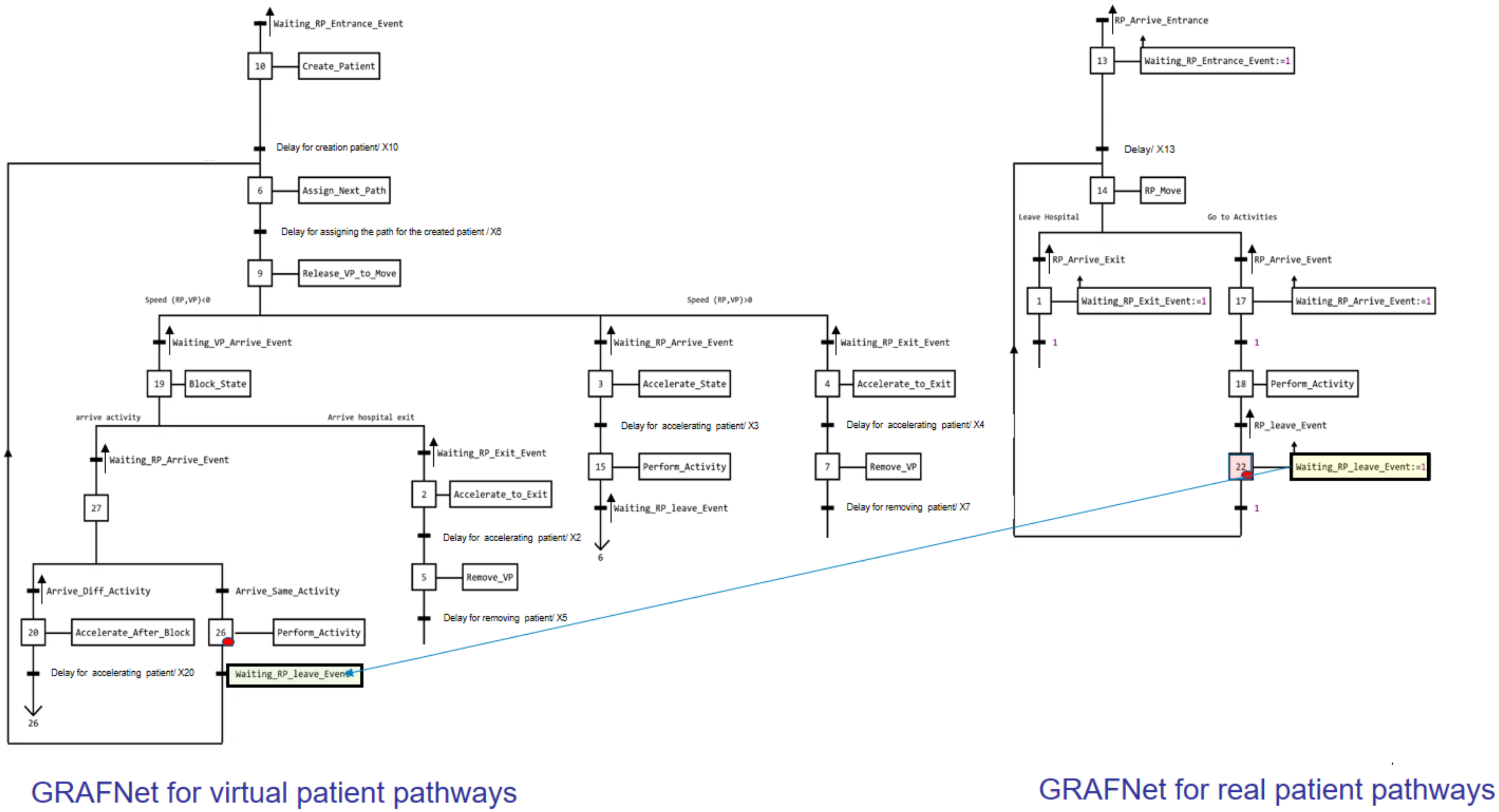


Figure 6.14: Synchronization between the virtual GRAFNet patient pathway and the real GRAFNet patient pathway: Step 7

Contents

Résumé

HospiT'Win : conception d'un double numérique basé sur la simulation à événements discrets pour le suivi en temps réel et la prédiction à court terme des parcours des patients à l'hôpital

L'hôpital est un lieu de travail exigeant qui nécessite une prise en charge en temps réel du patient et une interaction humaine élevée au niveau des ressources (médecin, infirmières, etc.), de l'emplacement (salle d'examen, salle d'opération, etc.), des processus (parcours de soins et de santé) et au niveau des usagers (patients). Concevoir et manager un établissement de santé peut être difficile. D'une part, parce que les services peuvent être des ressources critiques, et présenter une grande variété et une grande variabilité. D'autre part, l'adaptation à une demande de soins partiellement imprévisible peut se révéler très complexe.

Pour augmenter l'efficacité et la qualité des soins tout en limitant les coûts, un hôpital a donc besoin d'outils d'aide à la décision. Ils peuvent être basés sur des méthodes et des outils d'ingénierie organisationnelle afin de suivre l'état des services en temps réel, de prédire leur comportement dans un futur proche, et d'améliorer les processus. Parmi les méthodes et outils disponibles, la simulation à événements discrets (SED) permet de mieux comprendre le comportement des processus opérationnels et d'évaluer leur performance en simulant une chaîne d'événements qui se produisent dans le temps. Cependant, bien que les logiciels de SED disposent de multiples fonctionnalités, ils se limitent principalement à la construction de modèles de simulation « hors ligne » qui ne sont donc pas connectés au monde réel en temps réel. Par conséquent, les modèles ne sont pas adaptés pour récupérer l'état courant d'une organisation à un instant précis, ils ne peuvent donc pas être considérés comme des doubles numériques (DN). De plus, les simulations démarrent dans un état « vide » et « inactif », qui peut être différent de l'état réel, ce qui peut entrainer un biais dans les rapports statistiques à la fin de la simulation. Notre travail de recherche propose une approche de double numérique à base d'un simulateur à événement discret. Notre double numérique fournit une représentation virtuelle et en temps réel qui est synchronisée avec les ressources physiques et/ou les activités des processus. Il est basé sur des modèles de SED qui sont utilisés pour (1) le suivi en temps réel et en ligne des parcours des patients, et (2) la prédiction hors ligne du futur proche face à un comportement inattendu ou à des situations imprévisibles.

L'objectif principal de cette thèse est de fournir un cadre pour la construction d'un double numérique du parcours des patients que des professionnels de santé et des décideurs pourraient utiliser comme outil d'aide à la décision. Plusieurs problématiques spécifiques sont également abordées : l'initialisation des modèles de SED sur l'état courant, la synchronisation en temps réel avec le monde réel, et la connexion entre le modèle de suivi et le modèle de prédiction. Comme preuve de concept, nous proposons des expérimentations basées sur un émulateur d'un service hospitalier connecté à un double numérique développé suivant notre approche.

Mots-clés : Double numérique, Simulation en ligne, Simulation hors ligne, Simulation d'événements discrets, Parcours des patients à l'hôpital, Simulation prédictive, Données en temps réel, Internet des objets

---

Abstract

HospiT'Win: Designing a Discrete Event Simulation-Based Digital Twin for Real-Time Monitoring and Near-Future Prediction of Patient Pathways in the Hospital

Hospitals are demanding workplaces that have real-time services and require extensive human interaction at the resource level (doctors, nurses, etc.), the location level (exam rooms, operating rooms, etc.), the process level (pathways), and the user level (patients). Designing and managing such a health care facility can be inherently challenging due to the critical nature of the services and their wide variety and variability, as well as the difficult adjustment to a partially unforeseeable demand of care.

To increase the efficiency and the quality of care while curbing healthcare costs, a hospital requires decision-making support tools. These can be based on organizational engineering methods and tools for monitoring the current state of the organization in real time, for predicting its behavior in the near future, and for improving its processes. Among the tools, discrete event simulation (DES) is able to model the operational process behavior and to assess performance by simulating a chain of events that occur over time. However, despite important features provided by DES software tools, they are often limited to building "offline" simulation models that are not connected to the real world in real time. These simulation models may not be suitable for retrieving the current state of the organization, and they cannot be considered a "Digital Twin". Furthermore, these simulations start with an "empty" and "idle" state, which can be different from the real-world state, and imply a bias in the statistics reports at the end of the simulation run.

This research work deals with a DES-based Digital Twin (DT) approach. It is based on DES models which are used (1) for real-time and online monitoring of patient pathways, and (2) for near-future offline prediction when facing unexpected behavior or unpredictable situations. The major goal of this research is to provide a framework for building a Digital Twin of patient pathways that health care practitioners and decision makers can use as a decision support tool. Some specific issues are also addressed: initialization of the DES models, real-time synchronization with the real world, and the connection between monitoring and prediction models. As a proof of concept, experiments are carried out using an emulator of a hospital service that is connected to a Digital Twin that follows our approach.

Keywords: Digital Twin, Online Simulation, Offline Simulation, Discrete Event Simulation, Patient Pathways in Hospital, Predictive Simulation, Real Time Data, Internet of Things