9th International Workshop on Innovative Simulation for Healthcare
17th International Multidisciplinary Modeling & Simulation Multiconference

ISSN 2724-0371 ISBN 978-88-85741-50-8 © 2020 The Authors.
DOI: 10.46354/i3m.2020.iwish.012

HospiT'Win: a digital twin framework for patients' pathways real-time monitoring and hospital organizational resilience capacity enhancement

Abdallah Karakra[1,*], Elyes Lamine[1,2], Franck Fontanili[1] and Jacques Lamothe[1]

[1]Industrial Engineering Department-University of Toulouse-IMT Mines Albi, Route de Teillet 81000 Albi, France
[2]ISIS, Institut National Universitaire Champollion - Toulouse University,Rue Firmin-Oulès, 81104 Castres, France

[*]Corresponding author. Email address: abdallah.karakra@mines-albi.fr

Abstract

The recent challenges presented by the Coronavirus pandemic (COVID-19) are examples for the need to improve the soundness and resilience of hospital management. Digital Twin technique seems to be a relevant means to attend these needs. It consists of virtual representations of real assets and/or processes that are used to understand, predict and optimize their operation and efficiency. The present work sets out to investigate the usefulness of this technique for hospital management and points out the process of developing a digital twin framework dedicated to real-time monitoring of patients' pathways and predicting their near future. It aims to handle irregular, unusual and unexpected behaviors that may happen in hospitals and helps to make the right decision to mitigate the unpredictability situation. Different issues related to the way of developing, initializing and synchronizing the digital twin are discussed in this paper.

Keywords: Digital Twin; patients' pathways; Online Simulation; Real Time Monitoring; Predictive Simulation.

1. Introduction

Nowadays, ensuring business resilience has proven to be increasingly challenging since the business environment is becoming more and more unpredictable. In fact, for several years to now, many exceptional events have reminded us of the vulnerability of companies to major disasters or losses, whose consequences for people, assets and the environment are sometimes disastrous; such as the recent health crisis that the world is currently living in.

Organizations are requiring new methods and tools to cope or recover their performance and efficiency after disruptive events. The hospital is one of these organizations. Indeed, we have recently noticed this Coronavirus (COVID-19) pandemic. The hospital manager needs to manage the patients' pathways and their evolution over time in order to strengthen their resilience in the face of major risks.

At this stage, it is worth to notice that the quality and the relevance of the decisions that should be taken are strongly linked to the visibility that these decision-makers have on the given situation and its future development. Hence, monitoring the patients' pathways in real time is one of the most important aspects that deliver 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) (Karakra et al., 2019). In the industrial domain, there are different techniques used for monitoring the physical systems and for predicting their near future. Digital Twin (DT) approach is one of them and has gained much importance in recent years. It defined as (physical and/or virtual) machines or computer-based models that are simulating, emulating, mirroring, or "twinning" the life of a physical entity, which may be an object, a process, a human, or a human-related feature. It is more than a simple model or simulation (Barricelli et al., 2019). It seems to provide a valuable means to collect timely information that fully describes a potential or actual physical situation (Grieves and Vickers, 2017).

Although the objectives of DT are very ambitious, scientific research in the healthcare field compared to what is published in the manufacturing field is scarce and still at its early development stage (Barricelli et al., 2019). There is no methodological framework to structure and guide the design of a digital twin according to hospital decision maker needs.

Since this work sets out to investigate the usefulness of digital twin for hospital management and its practical considerations. It contributes to set the first foundations of a new framework called HospiT'Win. It figures out the process of building a DT for monitoring and predicting the patients' pathways in real time. This process includes the way of initializing the DT, synchronizing the DT twin with the real world, and then predicting their near future.

This paper is structured as follows: Section 2 provides a survey of the recent digital twin applications in the healthcare domain. The core contribution of this paper, namely HOSPIT'WIN approach, is presented in Section 3. We report on an evaluation of this approach in Section 4 before we conclude with lessons learned and an outlook to future research directions in Section 5.

2. State-of-the-Art

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

To date, DT has been successfully implemented in different industries, notably manufacturing, aerospace, defense, manufacturing and building construction. These applications differ on the level of their focused functional description. Some of them were applied on the product/component level (as to screen out unwanted product functionality and features), others focused on operation level (as for learning a new business practice or for assessing a new treatment), and others point up the process level (as for predicting the near future of the traffic) and/or the system level (as for providing early and timely insights into system behavior).

Currently, there are also different ongoing researches that illustrate the importance of applying the DT in the healthcare domain. (Martinez-Velazquez et al., 2019) suggest a DT platform of the human heart called "Cardio Twin". The main idea behind this twin could be summarized by collecting data from different sources. Such as, sensors, medical records, social networks from all. This collecting data is processed using different techniques to detect if the patient suffers from heart diseases such as Ischemic Heart Diseases (IHD) or a Stroke. The future work for this research is to extend the Cardio platform to help in the prevention of the mentioned diseases by reducing the risk factors associated with these diseases. Another DT of heart developed within the "Living Heart" project powered by the French software firm Dassault Systemes. It consists on 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 tool guide device design and treatment planning in cardiac diseases such as stenosis, regurgitation, or prolapse of the aortic, pulmonary, tricuspid, or mitral valve (Baillargeon et al., 2014).

Within a broader scope, the Virtual Physiological Human (VPH), an EU initiative project, aims to develop an integrated model of human physiology at multiple scales from the whole body through the organ, tissue, cell and molecular levels to the genomic level (Viceconti et al., 2008). VPH is intended to support the development of patient-specific computer models and their application in personalized and predictive healthcare (Kohl and Noble, 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 learnt automatically 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 suffers from heart diseases such as IHD. In a related matter, (Liu et al., 2019) propose a cloud-based framework for the 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 healthcare. Machine learning algorithms are implemented for fast simulations in order to predict crisis situation.

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 like a personal digital health related collection bag, carried by individual citizens throughout their lifetime capable of sustaining in a meaningful manner all collected information. This information is related to multilevel personal health data that is collected from heterogeneous data sources such as clinical data, genetic data, and medical sensor data from all(Kondylakis et al., 2015). (Barricelli et al., 2020) propose an extension to SmartFit which is a computational framework exploiting wearable sensors and internet applications. The extension is able to monitor a team of athletes, predict their condition during training and then suggest changes in behavior to increase performance and health conditions.

From healthcare organizational perspective, a recent work carried out by GE Healthcare company has implemented a predictive real-time platform in a new “command center" department for the Johns Hopkins Hospital in Baltimore (INFORMS, 2017). It aims at predicting 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 bed assignments or whether a unit needs assistance or an influx of patients coming into the hospital. Similar work has also been pursued by Siemens Healthineers company (Scharff, 2018) in which 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 lack of space by instantly evaluating various options before implementation of the right solution to transform care delivery. This DT is drawn upon workflow simulation and 3D computer model enabling the building of a dynamic and comprehensive model which integrate patient pathways, staff scheduling and movements. Similarly,(Augusto et al., 2018) propose a framework for modelling and simulation in the form of an offline DT to evaluate the performance of emergency units in the case of major crisis such as earthquake, tsunami, terrorist attack.

Table 1 provides a comparative study of these various DT applications according to their uses and their functional description level.

The present studies confirm the effectiveness of the digital twin healthcare (DHT) to pave the way for an efficient path for patients to access safe, effective new treatments and cares and to reduce costs. However, as far as we know, and as noticed in several other studies related to this topic (Barricelli et al., 2019), DTH projects are still partial and focus particularly on the virtual patient and well-being. Up to now, far too little attention has been paid to the role of DT for hospital management which is the main subject of this paper.

3. Proposed approach: HospiT’Win

Today, managing growing patient demand, decreasing of waiting times and delays, enhancing resilience to sustain required operations under both expected and unexpected conditions are common challenges that most hospitals are suffering (Karakra et al., 2018). To overcome theses challenges, we propose a digital twin of patients’ pathways for the hospital of the future called HospiT’Win. It is used to convey key information to decision makers in real-time. It is a high-fidelity and dynamic virtual representation of patients’ pathways inside the healthcare organization. It works based on Discrete Event Simulation (DES) which is a well-recognized method for modelling and analyzing health care services (Zeigler, 2014).

As illustrated in figure 1, HospiT’Win has three main components: the Real world, the virtual world, and the connection between them. In the real world, there are patients’ pathways that represent the core objective of this work. These pathways are considered as the processes that each patient follows from entry into a hospital or a medical unit until discharge. They are seen as timelines on which every event relating to treatment can be entered, including consultations, diagnosis, treatment, medication, and preparing for discharge from hospital (Dictionary, 2011).

The virtual world is a composite of a Digital Twin for monitoring (DTM), and a Digital Twin for Predicting (DTP). The cloud database has been used to bridge the gap between the two worlds. It allows the synchronization of states of the enacted digital twin with patients’ pathways. For example, each time an event is detected in these pathways, information about this event will be injected in the database in real time. For 3D representation and monitoring the patients’ pathways, DTM could be used for this purpose.

First of all, DTM will be initialized with the current state of the real patients’ pathways from the database, then it will be synchronized with the database to reflect any new event that may happen in these pathways.

For predicting the future, the DTP could be used for this purpose. Firstly, DTP will be initialized with the current state of the real patients’ pathways from the database. Then, it will be running at clock speed faster than the real world speed to anticipate the near future.

The predictive model in HospiT’Win is not a classical simulation model as in (Augusto et al., 2018). Actually, it is a half connected model. Which means this model is connected with patients’ pathways for continuously

Table 1. Comparing several DT projects dedicated for healthcare sector

		Functional description level				Main usage				
Digital Twin project	**Scope**	**Product**	**Operation**	**Process**	**System**	**Learning**	**Designing**	**Monitoring**	**Predicting**	**Preventing**
(Martinez-Velazquez et al., 2019)	Heart Diseases	○	○	○	●	○	○	●	○	●
(Baillargeon et al., 2014)	Heart Diseases	◐	●	○	○	●	●	○	●	○
(Viceconti et al., 2008), (Kohl and Noble, 2009)	VPH	○	●	○	●	●	○	○	○	●
(Ayache, 2019; Ayache et al., 2011)	Virtual patient	○	●	○	●	●	○	●	●	○
(El Saddik et al., 2019)	Health & well-being	○	○	○	●	○	○	●	●	●
(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

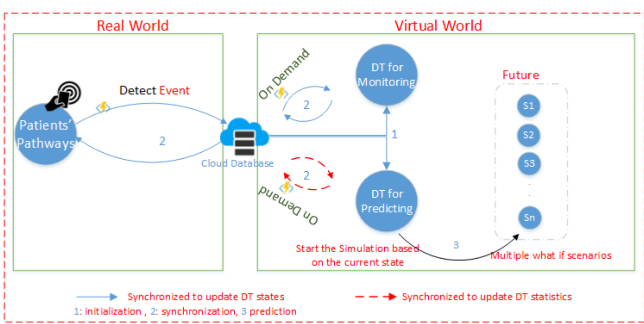


Figure 1. Proposed approach: HospiT'Win

updating its statistics and probabilities. Hence, in this approach, *initializing* and *synchronizing* the DTM and the DTP with real patients' pathways are two fundamental methods that guarantee the quality and the accuracy of the provided information and insights.

3.1. Initialisation

In this work, we consider the initialisation as the starting step before synchronizing the model with the real world. It means starting the model with a "non-empty or idle state". In our case, starting the model in the same state as the real patients' pathways in the hospital (Bergmann et al., 2011). Actually, this is different from the traditional simulation where the simulation model starts with an empty or idle state (Hanisch et al., 2005). In this approach, there are four related sets of data:(1) R represents a set of activities in the real hospital (2) V represents a set of activities in the virtual hospital Model (DTM or DTP) (3) IDsR represents a globally unique set that is used to keep track of the patient IDs in the real hospital Model (4) IDsV represents a globally unique set that is used to keep track of the patient IDs in the virtual Model. To clarify this approach, the following notations will be used:

- r represents a real world activity, where $r \in R$ and r={(activity name, current number of patients in this activity, IDsR)}.

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:             j ← (j + 1)                          ▷ Updated to read the ID of the next patient
8:         end while
9:         (V.v[i]).numberOfPatients ← (R.r[i]).numberOfPatients    ▷ Update the number of patients in activity v[i]
10:    end for
11:    DTMClock ← RealWorldClock                    ▷ Assign the DTM clock to the real world clock
12: end procedure
```

- | R |, return the number of activities in the set R. In algorithm 1, to remove the confusion R.length has been used as a synonym to the | R |.
- v represents a virtual world activity, where $v \in V$ and v={(activity name, current number of patients in this activity, IDsV)}.
- | V |, return the number of activities in the set V.

In this presented work, five activities have been considered; Waiting Line (WL), Registration Desk (RD), Waiting Room (WR), Exam Room 1 (ER1), and Exam Room 2 (ER2). Consider that in the real hospital at 14:30:00, there are three patients on the waiting line, one patient at the registration desk, two patients in the waiting room, one patient in exam room 1, and no patients in the exam room 2. The sets R,V, and IDs before running the DTM has the following values: R= {(WL,3,{123, 234, 345}), (RD, 1, {678}), (WR, 2,{490, 530}) ,(ER1, 1,{773 }), (ER2, 0,{})}, and V= {(WL, 0,{}), (RD, 0,{}), (WR, 0,{}) ,(ER1, 0,{}), (ER2, 0,{})}.

As can be noticed, all of the activities in set V are empty since the DTM is off and not running till now. While the sets R and IDsR are not empty, the set R has the same number of patients at each activity in the real hospital, and the set IDsR contains the IDs of those patients. For example, there are three patients on the waiting line, and the IDs of those patients are (123, 234, 345), and so on. Consider the state of the real hospital at this moment, it does not change, and we need to run the DTM. In this case, the values of the set V had to update from empty to have the same values of set R with the same IDs. Such as; V= {(WL,3,{123, 234, 345}), (RD, 1, {678}), (WR, 2,{490, 530}) ,(ER1, 1,{773 }), (ER2, 0,{})}. Algorithm 1 shows the process of initializing the DTM to have the same state as the real world (real patients' pathways) state.

Technically, one can use a database table to implement the two sets R and IDs. Such as, the header of this table could be {TableID: int, PatientIDs : text, PatientLocation: int, ...}. The first step of algorithm 1 is to call a Reset procedure. This Procedure is responsible for pulling the patient information from the database and creating virtual patients in the DTM model corresponding to their information that was pulled from the database, such as their IDs and their locations at each activity. The loop in the algorithm 1 in line 5 is responsible for creating virtual patients in DTM corresponding to their real twins in the real hospital by taking in consideration their IDs and the current activities where they exist. Line 11 in the algorithm is responsible for initializing the DTM clock with the real world clock.

3.2. Synchronisation

In this work, we consider the synchronization as the second step after the initialization where the virtual patients' pathways must have the same state as the real patients' pathways at each point of time (at each detected event). The quality and reliability of DTM depends on the quality of synchronizing the DTM with the real world. In other words, DTM must have the same state as the real world in the real time. Based on the defined sets (R and V) in section 3.1. The set V must have the same values as the set R at each detected event. For example, each time the set R is updated, the set V must be updated too, at the same time. In the synchronization approach, there are three cases which must be considered: (1) new patient enters to the hospital (2) the patient moves from one activity to another (3) the patient leaves the hospital.

When the real patient enters the hospital, a virtual patient must be created at the entrance door of the DTM (the set V) corresponding to this patient (algorithm 2, line 4). Also, a random next destination will be assigned to this patient to keep walking, even if this destination is wrong. When the real patient arrives at the next destination, this virtual patient will be synchronized to have the same location as the real patient (algorithm 2, line 5). In case the real patient leaves the hospital, the virtual patient must be removed from the DTM (algorithm 2, line 7). The problem is when the real patient moves between the activities (algorithm 2, line 9). In case the DTM receives an event; the real patient has started to move from activity called X to wherever. The virtual patient corresponding to this moving real patient must start to move. To synchronize those two patients, the synchronization point will be determined when the real patient arrives at the next activity. Due to the fact, this DTM depends on Discrete Events (DE). For this reason, the synchronization will happen only in one case when DTM receives an event from the real hospital. In this synchronization approach, three different scenarios are discovered when the patient moves between the hospital's activities:(1) Real patient is faster than virtual patient (2) Real patient is slower than virtual patient (3) Wrong destination/path: the virtual patient does not know the next destination until the real patient arrives. For example, the virtual patient goes to activity called X (wrong activity), while the real patient goes to the activity called Y.

Algorithm 2 shows the process of synchronizing virtual patients' pathways with real patients' pathways. It illustrates an overview of the main functions that needed to be used. The implementation of these functions may differ from one programmer to another. Different shortcuts have been used in this algorithm such as: VP stands for virtual patient, CreateVP stands for create virtual patient, RemoveVP stands for remove the virtual patient and SynchVPLocation means synchronizing the location of the virtual patient to have the same location of the real patient.

Algorithm 2 Synchronization

```
Require: EventType                          ▷ Enter, Exit, or Move
 1: procedure StartSynch()
 2:     EventType ← waitRealWorldEvent()
 3:     IF EventType= Enter Then
 4:         VP ← CreateVP(HospitalEntrance, RP_ID)
 5:         AssignNextDistPath(VP)
 6:     ELSE IF EventType= Exit Then
 7:         RemoveVP (VP)
 8:     ELSE
 9:         SynchVPLocation (VP)
10:     END IF
11: end procedure
```

For the SynchVPLocation procedure, there are two different approaches that could be used to do the synchronization: *removing and creating patient approach*, *accelerate patient approach*. By applying the first approach, when the real patient arrives at activity called 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 created again at activity X with the same ID. This case represents the scenario when the real patient is faster than the virtual patient. In case the virtual patient arrives at the activity X while the real patient is still walking. In this case, the virtual patient must be blocked from starting the activity until the real patient arrives at the same activity. This case represents the scenario when the real patient is slower than the virtual patient. The third scenario, if the real patient and the virtual patient went to two different activities. In this case, the virtual patient must be released to the same location of the real patient. This could be solved by removing the virtual patient from the model and creating him/her in the same location as the real patient. The problem in this approach is removing and creating the patient leads to loss of all the statistics related to this patient. Also, the model maybe crashed due to many calls for the removal and creating procedures for the same patient. For this reason, this paper recommends the usage of *accelerate the patient* approach to keep the statistics of each patient and to save the model from crashed. In this case, if the real patient is faster than the virtual patient, the virtual patient must be accelerated to the same location as the real patient. If the virtual patient is faster, the virtual patient had to block from starting the activity until the real patient arrives at the same activity. In case of a different direction, the virtual patient must be released to the same location of the real patient. This could be solved by accelerating the virtual patient to the same location of the real patient.

4. Experimental Approach

4.1. Experimental Platform

Since we are in the research phase, we believe that connecting the proposed approach with a real hospital at this phase is too risky due to many reasons. First, the need to validate the proposed models before connecting with the real hospital. For example, validate their structures and behaviors. Second, validate the functionality of DTM and the returned control feedback from DTP. Third, the need for not disrupting the hospital activities during the period of testing the proposed approach. For these reasons, we designed an emulator to be an experimental platform that mimics a fictional hospital. This hospital behaves as the real one. To validate the monitoring and the prediction features for the illustrating approach, we connected the DTM with the developed emulator. For example, each time an event is detected in the emulator, this event is pushed into the DTM. WITNESS simulation tool has been used to develop this emulator, and FlexSim discrete event simulation tool has been used to develop the DTM and DTP.

In this work, two types of models have been explained in detail. These models are considered as the first step before developing the HospiT'Win. These models are Replay Model and prospective Model. The future work of this research is to disconnect this DTM from the emulator, customize it and connect it with the real hospital.

4.2. Case study

This case study represents data that came from a fictional hospital. Even if the scenarios in this study are realistic and represent a real world hospital, the data is not. In this experiment, two types of data have been collected as represented in the set D. D= {Patients_IDs, Virtual_Sensors_IDs}. This data represents the ID of the patient and the ID of the sensor that detected this patient. Those sensors are called virtual sensors because they exist in the emulator and not in the real world. In this experiment, we know the location of each patient from the sensor ID that detected this patient. Each sensor is located in a fixed position in the emulator. More details about the location of these sensors exist in the section 4.3.

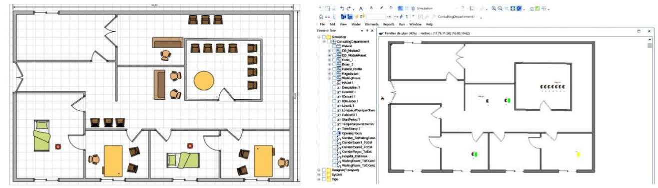
4.3. Emulator: Fake Hospital

To achieve the objective of this research without interrupting, affecting, or disrupting the daily activities of the hospital, WITNESS simulation tool has been used as an emulator to build the fake hospital. First, the emulator has been fed by the architectural layout of the hospital. Second, fixed resources and human resources have been located inside the emulator. To reflect the reality, stochastic distribution has been used to generate patients in different time slots, to move patients with different speed (fast, slow, and average) and in different pathways, and to set the duration of each activity.

In the emulator there are different types of patients coming from source to sink. Source has been used to generate patients inside the model (called hospital entrance), sink has been used to remove patients from the model (called hospital exit). During the journey of the patients from the hospital entrance to the hospital exit, there are different discrete points where the patients pass through. These discrete points are considered as fakes sensors. For each activity inside the hospital, there are two sensors; one detects the patient when he/she arrives at the activity, and the second one detects the patient when he/she goes outside the activity. At each sensor, there is a database query that writes the event information in the database. This information represents the ID of the patient detected by this sensor, the ID of the sensor that detected this patient, and the detection time (time stamp for this event). Figure 2, illustrates the developed emulator.

4.4. Replay Model and Prospective Model

To start developing DTM and DTP, offline models are essential. The offline models will be used in the validation phase before the connection with the real world.



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

Figure 2. The architectural layout and the emulator for the fake hospital

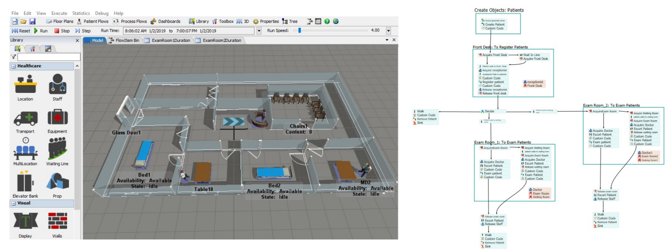


Figure 3. Replay Model (3D design view and process flow)

To do the validation, historical data is needed. To generate this data, the emulator has been run to generate a log file of events for two months. This file contains the following set of data {patient ID, Timestamp, Event Description}.

Replay model is an offline virtual representation model that used to replay log file events.FlexSim discrete event simulation tool is used to build this model. Actually, this model must behave nearly the same as the log file. For example, patients come to the hospital corresponding to the arrival time that exists in the log file, and the duration time for each activity for each patient must be the same duration time in the log file, In addition to all of that, the virtual patient must follow the same pathways that were discovered in the log file, and so on. Figure 3 illustrates the replay model.

The main reason for building the replay model is to validate the structure and the behavior of the model before connecting it with the real world. Actually, it is used to check if the actual execution of this model as recorded in the event log.

Prospective Model is a virtual representation model used to simulate the process using distributions of random variables (time, walking speed, rules, etc.) rather than using a log file. This model runs based on the patients' profile. In other words, this model is based on the static mathematical distribution laws that were discovered from the log file. One of the main differences between the replay model and the prospective model is that the execution of the replay model must be as recorded in the event log file, while the execution of the prospective model is based on the statistical mathematical functions that were discovered from the log file. The second difference is the replay model is used as a reference model to validate the prospective model.

This point has been explained in section 4.5. For the 3D design and the process flow, the replay model and the prospective model are the same. The third and the important difference between the two models, the replay model controlled by a specific period of time; the start point of the replay model is the start date in the log file, and the model stays running until reach the last date in the log file (works for two months in this experiment), while the prospective model is not limited to any date and time, the model could be run from the past and accelerated to the near future. Figure 4, shows the time line for the replay model and the prospective model.

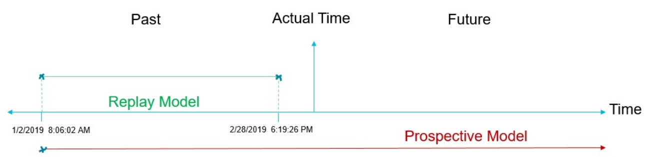
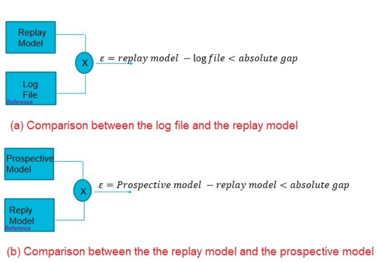


Figure 4. Time line: the difference between the replay model and the prospective model

4.5. Validation Process of the Replay Model and Prospective Model

To validate the replay model, a log file has been used as a reference. Based on a common selected set of indicators, the comparison between the log file and the replay model has been done as illustrated in figure 5 (a). To validate the prospective model, the replay model has been used as a reference model, and the comparison between the replay model and the prospective model has been done based on a selected set of indicators as illustrated in figure 5 (b).



Indicators	Description
ind_1	Incoming patient at Hospital
ind_2	Outgoing patients from Hospital
ind_3	Incoming patient at activity i
ind_4	Outgoing patients from activity i
ind_5	Min duration time for activity i
ind_6	max duration time for activity i
ind_7	Patients in process for 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

(c) Indicators used for validation

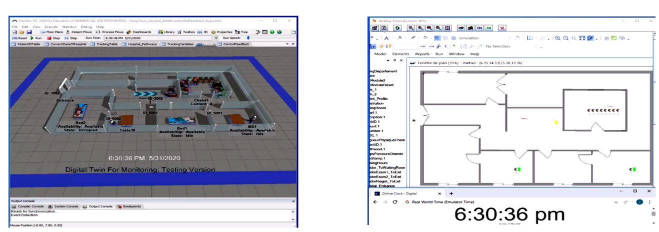
Figure 5. Validation

Figure 5 (c) demonstrated the different indicators selected by the authors to validate the models. These indicators could be different from one hospital to another. The hospital manager, including other hospital staff can participate in the selecting process for these indicators. In this work, we decided to say this model is valid if the difference between the reference model and the second model has an epsilon value (absolute gap) less than this range [-0.05, +0.05]. For example, if the incoming number of patients in the replay model in two months is 962, and the incoming number of patients in the prospective model for the same period is 942. In this paper, this indicator is valid because the difference between the incoming patients in the two models is in this range [913.9, 1010.1] which is acceptable compared with the predefined threshold range. Algorithm 3 shows the process of validation that was used in this experiment.

In algorithm 3, each indicator must pass the test to say this model is valid. If one or more indicators did not pass the test, that means this model is not valid except if the committee decides to say this model is valid for whatever reasons. In this experiment all of the indicators have been validated and passed the test.

4.6. Transformation Process: DT for monitoring and DT for Predicting

After validating the Replay model and the prospective model, now is the time for connecting these models with the real world. For this reason, the Replay model must be transformed to the monitoring model and the Prospective model must be transformed to the predictive model. To do this transformation, the replay model must be connected to the emulator instead of replaying the log file. In this experiment, the replay model listening to the events came from the emulator and reflected these events inside the model at the same time. For example, if there is a patient generated by the emulator at the entrance, the replay model must create a virtual patient at the entrance. If the patient goes to the registration desk in the emulator, the virtual patient must go to the registration desk in the replay model. The replay model in this case works based on the event received from the emulator in real time. We can call this model *DTM* instead of the replay model. To reflect the reality. The time of the DTM has to be synchronized with the time of the emulator, so that the DTM and the emulator have the same time (hours: minutes: seconds date). All of the events that came from the emulator to the DTM are kept and saved in the cloud database tables, so that we can use these events in the DTP. Figure 6 shows the DTM connected with the emulator.



(a) Digital Twin for Monitoring at 6:30:36 PM   (b) Emulator: Fictional Hospital at 6:30:36 pm

Figure 6. Digital Twin For Monitoring

Algorithm 3 Validation Process

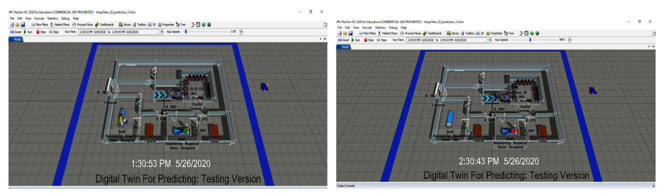
```
Require: maxRange, minRange, passTestCounter, indicatorsLength
function IsValid(IndicatorsList, ReferenceModel, Model)
    passTestCounter ← 0                                   ▷ Number of indicators pass the validation test
    indicatorsLength ← IndicatorsList.length              ▷ Number of indicators to validate
    for each indicator ∈ IndicatorsList do                ▷ To cover all of the indicators in the list
        maxRange ← ReferenceModel.value + (0.05 ∗ ReferenceModel.value)
        minRange ← ReferenceModel.value − (0.05 ∗ ReferenceModel.value)
        IF Model.value< maxRang and Model.value > minRange
            passTestCounter ← (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 figure 6 (a), DTM has the same state and behavior as the emulator in figure 6 (b). There are no patients on the waiting line and at the registration desk. Also, the number of patients in the waiting room is eight, and each room of the exam rooms has one patient. Furthermore, the emulator and the DTM have the same time 6:30:36 PM 5/31/2020 and 3:22:14 PM 4/24/2020.

For predicting the future, the prospective model works based on a static mathematical distribution that was discovered from the log file (static data from the previous period), as mentioned in section 4.4. This makes the model not dynamic and could not reflect the current situation since it works based on historical data. To solve this issue, the static mathematical distribution transformed into the dynamic mathematical distribution and in real time. For example, each time the virtual patient in DTM finished the activity, the mathematical distribution of the prospective model must be updated to consider this activity. Instead of using just historical data, we fed the prospective model with real time data. To do this, an empirical distribution has been used. In this configuration, the prospective model is transformed into a *DTP*.

4.7. Switching from DTM to DTP

In this work, to anticipate the near future for the patients' pathways, real time empirical distribution has been used. The process starts from the DTM. For example, the DTM starts monitoring the patients' pathways. In case an unexpected event is detected by this model, the DTP will run. The prediction steps are summarized as the following: (1) initialize the DTP with the current state of the emulator from the cloud database (2) synchronize the clock of the DTP to be the same clock for the emulator (3) run the DTP at a speed faster than the real world clock.



(a) Current state of the hospital at 1:30:53 PM (b) Predicted state of the hospital after one hour

Figure 7. Initializing and running the DTP to predict the next coming hour

Figure 7 illustrates an example for running the DTP, in figure 7 (a) At 1:30:53 PM, the DTP has been initialized with the current state of the hospital. In figure 7 (b) The DTP has been run based on the current state to anticipate the state of the hospital after one hour. For more clarification, the following sets of data represent the current state of the hospital and the anticipated states after one hour. Each set represents a set of activities with the corresponding number of patients at each activity: Hospital states at 1:30:53 PM = {(WL,0), (RD, 1), (WR, 1), (ER1, 1), (ER2, 1)}. Predicted states after one hour = {(WL,1), (RD, 1), (WR, 1),(ER1, 1), (ER2, 0)}.

5. Conclusion and Future Work

Recent developments in the field of Digital Twin Healthcare have led to a renewed interest in modelling and simulation methods to help optimize healthcare delivery services. On this basis, we have investigated this emergent technique for monitoring the patients' pathways in real time and then for predicting their near future. In this paper, we have shown the process of developing our digital twin framework named HospiT'Win based on the use of a discrete event simulation method. An experimental platform has been used to validate this process. In addition, this paper has discussed the way of initializing and synchronizing the digital twin for monitoring (DTM) and the digital twin for predicting (DTP) with the real hospital. Future research on this work aims to extend this experiment to use a local medical unit instead of an emulator, and to explore if there are any further issues that could be found and need to be solved.

References

Augusto, V., Murgier, M., and Viallon, A. (2018). A modelling and simulation framework for intelligent control of emergency units in the case of major crisis. In *2018 Winter Simulation Conference (WSC)*, pages 2495–2506. IEEE.

Ayache, N. (2019). AI and Healthcare: towards a Digital Twin? In *MCA 2019 - 5th International Symposium on Multidiscplinary Computational Anatomy*, Fukuoka, Japan.

Ayache, N., Clatz, O., Delingette, H., Malandain, G., Pennec, X., and Sermesant, M. (2011). Towards a personalized digital patient for diagnosis and therapy guided by image. *Medecine Sciences: M/S*, 27(2):208–213.

Baillargeon, B., Rebelo, N., Fox, D. D., Taylor, R. L., and Kuhl, E. (2014). The Living Heart Project: A robust and integrative simulator for human heart function. *European Journal of Mechanics - A/Solids*, 48:38–47.

Barricelli, B. R., Casiraghi, E., and Fogli, D. (2019). A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, 7:167653–167671.

Barricelli, B. R., Casiraghi, E., Gliozzo, J., Petrini, A., and Valtolina, S. (2020). Human digital twin for fitness management. *IEEE Access*, 8:26637–26664.

Bergmann, S., Stelzer, S., and Straßburger, S. (2011). Initialization of simulation models using cmsd. In *Proceedings of the 2011 Winter Simulation Conference (WSC)*, pages 2223–2234. IEEE.

Dictionary, S. M. (2011). Patient pathway | definition of patient pathway by medical dictionary. https://medical-dictionary.thefreedictionary.com/patient+pathway.

El Saddik, A., Badawi, H., Velazquez, R. A. M., Laamarti, F., Diaz, R. G., Bagaria, N., and Arteaga-Falconi, J. S. (2019). Dtwins: A digital twins ecosystem for health and well-being. *MMTC Communications-Frontiers*.

Grieves, M. and Vickers, J. (2017). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary perspectives on complex systems*, pages 85–113. Springer.

Hanisch, A., Tolujew, J., and Schulze, T. (2005). Initialization of online simulation models. In *Proceedings of the Winter Simulation Conference, 2005.*, pages 9–pp. IEEE.

Huang, Z., Dong, W., Ji, L., and Duan, H. (2016). Predictive monitoring of clinical pathways. *Expert Systems with Applications*, 56:227–241.

INFORMS (2017). Johns hopkins hospital opens capacity command center. http://analytics-magazine.org/johns-hopkins-hospital-opens-capacity-command-center/. (Accessed on 06/21/2020).

Karakra, A., Fontanili, F., Lamine, E., and Lamothe, J. (2019). Hospit'win: A predictive simulation-based digital twin for patients pathways in hospital. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pages 1–4. IEEE.

Karakra, A., Fontanili, F., Lamine, E., Lamothe, J., and Taweel, A. (2018). Pervasive computing integrated discrete event simulation for a hospital digital twin. In *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–6. IEEE.

Kohl, P. and Noble, D. (2009). Systems biology and the virtual physiological human. *Molecular Systems Biology*, 5(1).

Kondylakis, H., Spanakis, E. G., Sfakianakis, S., Sakkalis, V., Tsiknakis, M., Marias, K., Zhao, X., Yu, H. Q., and Dong, F. (2015). 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)*, pages 1397–1400. IEEE.

Liu, Y., Zhang, L., Yang, Y., Zhou, L., Ren, L., Wang, F., Liu, R., Pang, Z., and Deen, M. J. (2019). A novel cloud-based framework for the elderly healthcare services using digital twin. *IEEE Access*, 7:49088–49101.

Madni, A. M., Madni, C. C., and Lucero, S. D. (2019). Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems*, 7(1):7.

Martinez-Velazquez, R., Gamez, R., and El Saddik, A. (2019). Cardio twin: A digital twin of the human heart running on the edge. In *2019 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, pages 1–6. IEEE.

Roland Rosen, Stefan Boschert, A. S. (2018). Next generation digital twin. *atp magazin*, 60(10):86–96.

Scharff, S. (2018). From digital twin to improved patient experience - siemens healthineers global. https://www.siemens-healthineers.com/news/mso-digital-twin-mater.html.

Viceconti, M., Clapworthy, G., and Jan, S. V. S. (2008). The virtual physiological human—a european initiative for in silico human modelling—. *The journal of physiological sciences*, pages 0810200082–0810200082.

Zeigler, B. P. (2014). The role of modeling and simulation in coordination of health care. In *2014 4th International Conference On Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*, pages 1–5. IEEE.