Towards the development of a Digital Twin for the improvement of cool chain operational quality

A KLM Cargo case study

J.A. Sijtsma

Delft University of Technology

TU Delft

AIRFRANCE / KLM
Martinair CARGO

Towards the development of a Digital Twin for the improvement of cool chain operational quality

A KLM Cargo case study

By

J.A. Sijtsma

Master Thesis

in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical, Maritime and Materials Engineering of Delft University of Technology
to be defended publicly on Thursday, February 23, 2023 at 10:00 AM

Student number:	5349664	
MSc track:	Multi-Machine Engineering	
Report number:	2023.MME.8765	
Supervisor:	Dr. ir. Y. Pang	
Thesis committee:	Prof. dr. R. R. Negenborn,	TU Delft committee Chair, 3mE
	A. Nicolet,	TU Delft committee member, 3mE
	B. Krol,	Company supervisor, KLM Cargo
Date:	Tuesday 14th February, 2023	

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Cover Image: Loading a Full Freight Boeing 747 through the nose cargo door, BRIX AFKL Cargo (2015)

Preface

You have just opened the report that covers my graduation research, which is the final academic work for the Master program Mechanical Engineering - Multi-Machine Engineering at the TU Delft. I have performed my research at KLM Cargo, which was an opportunity for which I am very grateful. The dynamic environment of KLM Cargo, while overwhelming at times, provided many opportunities to further develop my skills and more importantly learn new ones. This graduation research has been a very rich experience full of challenges, new insights and most enjoyably a lot of laughs.

First of all, I would like to thank KLM Cargo for the opportunity to develop and conduct my own research project at the Business and Process Improvement department. Most importantly, I would like to thank my company supervisor Bart Krol for his patience and useful feedback and insights in order to improve my work. In general, I would also like to thank all the employees at BPI for their welcoming and open attitudes, as well as the other graduate interns with whom I have formed new friendships.

Furthermore, I would like to express my gratitude to the graduation committee for the guidance and supervision throughout the research process. Firstly, I would like to thank Prof. dr. R. R. Negenborn for the critical feedback and suggestions for the improvement of the project during our meetings. Of course, I would also like to thank my daily supervisor Dr. ir. Yusong Pang for his patience when my progress was slow, the very useful discussions we had on the research structure and specific contents and most of all his expertise and guidance throughout my research. Your style of supervision allowed me to truly be the project manager, from which I have learnt a great deal about myself and project management in general.

Above all, I would like to express my deepest appreciation to my girlfriend Els, whose unconditional support, encouragement and understanding throughout the past year has greatly helped me in the completion of this work. I would also really like to thank my friends and family for their continuous support. I would not be in the position I am right now if it was not for all of you.

For now, I hope you will enjoy reading this report.

J.A. Sijtsma
Rotterdam, February 2023

Executive summary

There is a rising demand for and reliance on pharmaceutical cool chains given the rise of the bio-pharmaceutical industry and tightening regulations around the world. However, several problems and challenges remain such as breaks in the cool chain and cool storage capacity constraints, ultimately impacting the operational quality of cool chains. One actor within the air freight industry facing such issues is KLM Cargo.

From the literature, it has been acknowledged that cool chain improvement may be focused on the information extraction and improved decision-making layers, while the application of the Digital Twin (DT) concept has been recognised as a potential for improvement. However, more research is needed on the application of DTs in logistics, especially the pharmaceutical cool chain. Furthermore, in order to quantify the operational quality of a cool chain, a novel metric has been introduced based on the Overall Equipment Effectiveness (OEE) methodology: the Overall Cool Chain Effectiveness (OCCE). The OCCE has been built up from three rates: the cool storage availability, on time performance and temperature adherence. Therefore, the research has been aimed at the following research question:

> To what extent can the pharmaceutical cool chain operational quality be improved through the development of a real-time decision-making methodology?

An answer to the main research question has been aimed to be achieved through the application of the DT concept in a digital system. Based on the studied literature, a DT has been defined as the additional functionalities offered by a digital system through interactions with a Physical Twin (PT) model representation of the physical system, while utilising automatic data connections. The proposed digital system has thus been built up from a PT simulation model, which represents the studied cool chain at KLM Cargo. The DT has been applied in the form of a decision-support module which offers additional functionalities to the system, compared to the PT.

From the studied cool chain process at KLM Cargo, it has been found that the cool storage facility is typically constrained in terms of capacity. Furthermore, the business ruling in place which determines whether freight receives cool storage, has been based on static ruling, depending on the transit time and Special Handling Code (SHC). Furthermore, although large quantities of data are generated in the system, this information is currently not utilised in the cool storage decision-making process. Moreover, a large amount of Key Performance Indicator (KPI) are in use at KLM Cargo in order to assess the operational performance of the system. However, these KPIs do not provide a coherent overview of the operational quality of the cool chain as a whole, while the temperature exposure on a Unit Load Device (ULD) level is currently not monitored. Therefore, it has been concluded that the studied pharmaceutical cool chain may benefit from the implementation of the DT concept through improved cool storage decision-making.

The PT model has been manually programmed in Python by means of Discrete Event Simulation (DES) and has been fully verified and partly validated with respect to performance data of the studied system, which has been noted as a limitation of the research. Similarly, the DT decision support module has been manually programmed in Python and loaded into the PT model. The studied system at KLM Cargo has thus been digitised through the PT model and has served as the baseline scenario in order to quantify the operational quality improvement by means of the OCCE metric. The DT module has been applied for the cool storage decision-making, therefore replacing the static business ruling with dynamic real-time business ruling based on the current system state. In specific, the dynamic real-time business ruling has been based on the expected exposure of a ULD.

From the PT model output in the digital system, it has been concluded that the operational quality of the pharmaceutical cool chain at KLM Cargo can be improved by 3.80% through the application of

the DT concept for improved cool storage decision-making. The operational quality improvement has been primarily achieved through the increase of the CRT cool storage availability increase. Furthermore, it has been found that during the critical periods of spring and summer, the operational quality improvement was most significant with a 6.39% increase, while overall the average exposure per ULD has decreased by 2.05%. Finally, it has been found to be possible to derive temperature profile plots for each unique ULD, with currently available data from the studied system, which could provide a method for improving the monitoring of temperature sensitive freight.

For further research, it has been recommended to consider the actual automatic data connections to and from the physical system and the digital system, in order to achieve an actual DT of a pharmaceutical cool chain. Besides, it has been recommended to consider the improvement of the improved decision-making algorithm for further operational quality improvements.

Contents

Acronyms

ABMS Agent-Based Modelling and Simulation. 16, 17
AFKLMP Cargo Air France KLM Martinair Cargo. 2, 3, 21, 22, 24
AGV Automated Guided Vehicle. 27
ATA Actual Time of Arrival. 27, 32, 33
AWB Air Waybill. 26, 27, 31, 33, 37, 38, 43–45, 49

BPI Business and Process Improvement. 2

CDG Charles-de-Gaulle. 21, 24
CEIV Pharma Center of Excellence for Independent Validators in Pharmaceutical Logistics. 2

DES Discrete Event Simulation. ii, x, 4, 17, 23, 41, 42, 45, 47–52, 54, 59, 61, 78
DT Digital Twin. ii, iii, viii–x, 5, 10, 11, 14, 15, 17–20, 23, 34, 42, 45, 48, 50, 66, 67, 70–79

EM Expectation-Maximisation. 13, 38
ETV Electric Transport Vehicle. viii, 29, 30, 49, 50, 56
EU European Union. 2

FAP Flown As Planned. 33, 44, 50
FTK Freight Tonne Kilometre. 21

GDP Good Distribution Practice. 2
GHA Ground Handling Agent. 8, 21

IATA International Air Transport Association. 2
ICA Intercontinental. 21, 25

KLM Koninklijke Luchtvaart Maatschappij. ii, viii, 2–5, 7, 20–28, 30–33, 36–38, 40, 42, 44, 79
kNN k-Nearest Neighbours. 4, 13, 38, 39
KPI Key Performance Indicator. ii, 3, 24, 31–34, 42–45, 50, 52, 55, 56, 58–60, 71, 74, 75, 77, 78

LTB Lower Temperature Bound. 44, 50, 68, 72

M-ULD Mixed-ULD. 25
MAR Missing at Random. 12, 13
MAS Multi-Agent Systems. 16, 17
MCAR Missing Completely at Random. 12, 13, 38
MICE Multiple Imputations by Chained Equations. 13
MNAR Missing Not at Random. 12, 13
MTD Moving Truck Door. 27, 29, 31, 32

OCCE Overall Cool Chain Effectiveness. ii, viii, x, 15, 23, 43, 45, 46, 50, 52, 53, 55, 56, 61–67, 71, 74–78
OEE Overall Equipment Effectiveness. ii, 4, 14, 15, 23, 77

PCHS Pallet Container Handling System. 25, 27–34, 38, 41–43, 48–50, 54, 56, 68, 70–72, 74, 76
PT Physical Twin. ii, viii–x, 5, 19, 23, 42, 43, 45–48, 50, 52, 55–57, 61, 66–71, 76, 78, 79

RFID Radio Frequency Identification. 10

SHC Special Handling Code. ii, 22, 23, 26, 27, 31, 33, 39, 47, 49, 54, 68
SPL Schiphol. viii, 8, 21, 22, 24–26
STA Scheduled Time of Arrival. 27
STD Scheduled Time of Departure. 27, 29, 30, 32, 33, 42, 49, 57, 58, 63, 70

T-ULD Through-ULD. viii, ix, 25–30, 33, 35, 40, 41, 47, 50, 54, 56–60, 62, 63, 65, 68–75, 78
TOR Time out of Refrigeration. 33, 42, 44, 56, 58, 59

ULD Unit Load Device. ii, iii, viii, ix, 2, 22, 23, 25–34, 37–40, 42–50, 54, 55, 57, 59, 62, 63, 67–72, 74–76, 78
UTB Upper Temperature Bound. 44, 50, 68, 72

WHO World Health Organisation. 2
WMS Warehouse Managment System. 3, 18, 29, 31, 35, 48, 49, 77, 79

List of Figures

List of Tables

1
Introduction

In the following chapter, an introduction to the research project has been given. Firstly, the research background has been provided in Section 1.1. Then, the research problem addressed in this work has been introduced in Section 1.2. Consequently, the research goal has been defined in Section 1.3 and the research scope has been discussed in Section 1.4. The chapter has been concluded by providing the research questions in Section 1.5 and an overview of the research structure and applied methodologies in Section 1.6.

1.1. Research background

The air freight logistics industry has become an increasingly important part of the modern global economy [1]. Annually, airlines transport over 52 million metric tons of cargo with a value equivalent to \$6.8 trillion [2]. The flow of air cargo includes products as diverse as cut flowers, pharmaceutical products, consumer electronics, perishable foods and medical diagnostic devices. Even though air freight shipments may account for less than 1% of global trade by volume, the total value accounts for 35% of all global shipments [3]. The air freight industry can be considered as a highly heterogeneous industry with a wide variety of major actors and traffic flows [1]. Nonetheless, three major actors in the air freight supply chain have been recognised: shippers, forwarders and carriers [4]. The shipper can be considered as the party who wants to ship cargo from one place to another. Consequently, the forwarder arranges the door-to-door transport of the shipment, handles the necessary documentation and possibly consolidates freight with the same destination into a single shipment. Finally, the carrier performs the airport-to-airport movement of freight. Although the industry generally exhibits turbulent behaviour with an uncertain future trajectory, it has been characterised by a relatively long-term growth rate for the past fifty years. Furthermore, despite the uncertainty, a demand increase has been anticipated for certain specialist products such as pharmaceuticals, cut flowers and medical diagnostic devices [1]. However, pharmaceuticals, fresh food and flowers are products characterised by the requirement of special handling conditions, especially regarding temperature. Therefore, these products are distributed through the so-called cool chain. Compared to supply chains for regular cargo, the cool chain includes all steps and facilities for storing, handling and transportation of perishable products, for which controlled temperature conditions must be maintained from the point of production to the point of sale [5]. The goal of an effective cool chain is to ensure a specific temperature range for specific products, such as fresh food products, medicine and vaccines. Therefore, besides the primary objectives of managing regular supply chains, cool chain management also aims at preserving the quality of products throughout the chain [6]. Depending on the temperature range, a cool chain may also be referred to in the literature as the cold chain if the supply chain has been designed to handle products under a low-temperature range such as 2°C to 8°C and below 0°C. However, in this work, any supply chain designed to handle temperature-sensitive goods has been referred to as a cool chain.

The primary product categories which can be distinguished that require an adequate cool chain are fresh and pharmaceuticals. Fresh products include different types of food such as fruit, vegetables, frozen meat, seafood, and dairy, as well as non-food products such as cut flowers. Pharmaceuticals include medicine, a wide range of vaccines and medical diagnostic devices. The demand for an adequate cool chain for each product category is driven by different developments. In terms of nutrition, the food industry is under significant strain to provide an adequate supply to an ever-growing human population. Alarmingly, however, it has been estimated that roughly one-third of food produced for human consumption is lost or wasted globally [7]. Among other reasons, food losses are generally attributed to natural decay, which can be accelerated by lacking or poor temperature management. In specific, unnecessary food losses tend to occur due to the fact that the actual temperature conditions during transport and storage often do not meet the optimal product-specific values [8]. Therefore, given a growing worldwide population, the challenge of nutritional supply could be significantly addressed through a reduction of losses throughout the supply chain. Although similar to fresh products, pharmaceutical shipments impose greater risks in terms of consumer health whenever environmental changes and fluctuations reduce the product quality [9]. This is due to the fact that many medicines and especially vaccines need to be preserved in a certain temperature range in order to remain effective. Whenever the effectiveness of a pharmaceutical product is compromised, it might ultimately put the receiving patient at risk. The reliance on and demand for a temperature-controlled cool chain for the pharmaceutical industry is actively driven by the expansion of the biopharmaceutical sector [10]. Compared to generic pharmaceutical products, biopharmaceuticals are based on biotechnology, which means that it is produced from naturally made protein, enzyme and antibody. Such biopharmaceuticals exhibit the advantage of low non-specific toxicity. However, disadvantages include the relatively high costs and sensitivity to the surroundings [11]. Due to the sensitivity of the products, biopharmaceutical is one of the major sectors in the pharmaceutical industry requiring temperature control. Furthermore, cool chain management is critical due to the high costs and thus significant shipment values of pharmaceuticals in general.

Despite the actively driven demand for effective cool chain management, several problems and challenges remain. As an example, an ideal cool chain provides the correct environmental conditions at all times and locations throughout a network. However, especially in the air freight industry, a network may contain many handovers of shipments that in principle constitute to breaks in the cool chain. During such breaks in the chain, the risk of temperature excursions and thus shipment losses significantly increase. Besides, Kartoglu and Milstien [12] have justly noted that a general illusion exists in the sense that cool chain problems are mainly encountered in developing countries. In fact, cool chain problems have been documented in all countries where temperature monitoring studies have been performed. Although such challenges have given rise to specialised active Unit Load Device (ULD)s, or refrigerated aircraft containers, there are rarely routine systems in place to provide consistent insight into cool chain performance and enable day-to-day performance management [13]. For the following work, it has been chosen to limit the research object to the pharmaceutical air freight cool chain only, which has been further elaborated in the research scope.

1.2. Problem definition

One actor within the air freight industry facing cool chain challenges is the Koninklijke Luchtvaart Maatschappij (KLM), or Royal Dutch Airlines in English, and the respective cargo division KLM Cargo at which a case study has been performed for this research project. KLM Cargo is the division of KLM which handles, prepares and finally offers the freight for air transportation to KLM, the carrier. Besides the arrangement of air transport of general cargo and valuables, the division has also invested significantly in a cool chain for both fresh and more importantly pharmaceutical shipments. The case study for this research project has been performed at the Business and Process Improvement (BPI) department, which focuses on the improvement of the cargo handling and transportation processes. The BPI department has an active and dedicated cool chain program aimed at the improvement of the cool chain processes, transparency and compliance. Naturally, pharmaceutical shipments are subject to strict regulations. In fact, regulations have been tightened in Europe, which has been followed by the United States [14]. In order to provide a common baseline from existing regulations and standards such as the European Union (EU) Good Distribution Practice (GDP) and the World Health Organisation (WHO) Annex 5, the International Air Transport Association (IATA) has established the Center of Excellence for Independent Validators in Pharmaceutical Logistics (CEIV Pharma) certification, for which re-certification is necessary every three years. Air France KLM Martinair Cargo (AFKLMP Cargo) has already received the CEIV Pharma certificate three times in a row, where the re-certification continues to demand sufficient quality management. Despite investments and certifications, the joint cargo division of the Air-France KLM Group Air France KLM Martinair Cargo (AFKLMP Cargo) has been losing market share in the pharmaceutical segment. In fact, between 2015 and 2018, the business share decreased from 27% to 15.8% respectively, while the total market has been growing. Customers have attributed the market share decline to, among other reasons, operational quality. The latter term is a broad concept and has therefore been further delineated for the studied research subject. In a general sense, transport logistics quality has been defined as the degree to which the performance of the freight transport operations across modes in the supply chains, meets stated service criteria [15]. The latter is a general notion of operational quality in transport logistics which has been extended to the cool chain. In essence, the operational quality of a cool chain has been defined as the performance of the system with respect to certain criteria or service levels. In other words, operational quality has been understood as the effectiveness of the system. Logically, the degree of operational quality or the effectiveness of the system should be quantifiable to allow for a proper assessment of the research object. For the cool chain studied in the case study, multiple KPIs are used to quantify the performance of the system. However, a single metric which represents the operational quality of the cool chain is currently not in use nor has it been defined.

In the current situation, the handling operations of pharmaceutical shipments are primarily driven through standardised handling process milestones while operational tasks are generated by a Warehouse Managment System (WMS), which thus lacks certain flexibility with regard to operational decision-making. Although supporting personnel has some ways of intervening in the cool chain process, it is generally done based on experience or intuition which may not lead to optimal results. Furthermore, the current monitoring capabilities of pharmaceutical freight at KLM Cargo remain limited. In this study, monitoring has been referred to as the process of continuously gathering logistics and programme information to verify whether the objectives are met [16]. As an example, only pharmaceutical shipments which are to be handled between 2°C and 8°C are monitored twice a day on a time and location basis. Although it is therefore possible to verify whether such a shipment is in a cool room at or before a given time according to the WMS, there is no insight into the actual state of the cargo and cool chain as a whole. The latter is of importance due to capacity limitations present in the cool cells used in the warehouse to temporarily store temperature-sensitive freight. In fact, it has not been uncommon for cool cells to be completely full, resulting in the inability to ensure the correct handling of pharmaceutical freight, which ultimately negatively influences the operational quality of the cool chain. However, expansion of the cool cells requires significant investments and may even not be possible given spatial arrangements in warehouses. Therefore, improved decision-making on the use of resources such as cool cell storage may provide an opportunity to improve the cool chain with the given infrastructure while allowing for proactive interventions in the process. In conclusion, the problem has been defined as follows:

Problem Statement

Currently, there is no capability in place for real-time decision-making in order to improve the cool chain operational quality based on the actual system state with the existing infrastructure.

1.3. Research goal

Based on the described problem definition, the goal of the research has been formulated as follows:

Research Goal

The research project has been aimed at the development of a real-time decision-making methodology in order to improve the operational quality of the pharmaceutical cool chain.

In principle, the goal of decision-making improvement entails the aim of utilising process and environmental data in order to determine, support and possibly improve the decision-making with regard to cool storage of pharmaceutical freight. In other words, the decision-making goal is to improve the decision-making with regard to whether freight should receive cool storage or not. Therefore, the research has been focused on the development of real-time decision-making based on the actual system state, for the improvement of the operational quality of pharmaceutical cool chains.

1.4. Research scope

Time is a substantial constraint in any research project. Therefore, in order to ensure the feasibility of this work, a certain scope has been set that simultaneously permits sufficient complexity in the studied system. Firstly, since the case study has been carried out at KLM Cargo, the scope has been limited to the domain in which this division operates. Although the latter can be considered self-evident, it should be recognised that the global cool chain from shipper to consignee is a rather extensive network involving many different parties. However, KLM Cargo only has the capability to intervene in the part of the cool chain for which the company bears responsibility. Therefore, the scope has been limited to this domain, which has been further elaborated in Chapter 3. Secondly, it has been decided to limit the scope to pharmaceutical shipments only, therefore not considering fresh or perishable shipment flows. The fresh and pharmaceutical products do not differ significantly from a technical perspective; the process steps are comparable and there is no differentiation in the equipment used to handle either type of shipment. However, the products within the shipments are inherently different and customers undoubtedly have a different and more stringent regulatory responsibility for pharmaceutical shipments. In principle, the latter is also true for KLM Cargo and for instance temperature violations are typically of greater concern for pharmaceutical customers, who also demand higher standards compared to fresh customers. Finally, it has been chosen to limit the scope to the truck-aircraft transit shipment flow only at the hub of KLM Cargo. With regard to the proposed methodology, actual implementation has not been achieved in this research project. Therefore, the automatic connection of data exchanges between the proposed digital system and the physical system and vice versa has not been included in the scope.

1.5. Research questions

The research has been structured according to a main research question along with several sub-questions:

Research Questions

To what extent can the pharmaceutical cool chain operational quality be improved through the development of a real-time decision-making methodology?

1. Considering the state of the art, how can cool chain management be improved?
2. What is the current state of a pharmaceutical air freight cool chain process, based on an applied case study?
3. How can a pharmaceutical air freight cool chain be modelled?
4. To what extent does the developed model effectively represent the research object, in terms of verification and validation?
5. How can the improved decision-making be implemented?
6. Which insights can be derived from the developed model?

1.6. Research methodology

In order to clarify the methodology and structure used for answering the presented research questions, the research framework has been visualised in a conceptual model, which can be seen in Figure 1.1. In order to obtain the research goal, several methodologies have been used. Firstly, the OEE has been applied in order to define a suitable effectiveness measure for the operational quality of the pharma cool chain. Although metrics such as the OEE have been defined in the literature in order to study the effectiveness of individual machines, production lines and even production facilities, a similar metric has not been found in the literature for a cool chain. Therefore, in order to quantify the improvement instated by the proposed methodology, the OEE methodology has been adapted to fit the needs for an effectiveness metric of a cool chain which can quantify changes in terms of operational quality. Secondly, as typically encountered in industry, k-Nearest Neighbours (kNN) missing data imputation has been applied in order to obtain an adequate data input for the process modelling. Thirdly, in order to quantify the extent to which the operational quality of a pharmaceutical air freight cool chain can be improved by the proposed methodology, Discrete Event Simulation (DES) modelling has been applied in order to experiment with the studied system. Finally, in order to achieve the research goal, a decision-making improvement method is required. For fresh cool chains, a significant area of research has been aimed at extracting the product state or quality remotely, in order to assist the decision-making process in such systems. Similarly, a separate area of research which has been under development is that of a Digital Twin (DT), which can be considered as a virtual copy of a real-world object through which additional functionalities may be offered. Interestingly, the concept of a DT has been applied in fresh cool chain studies [17, 18, 19, 20, 21, 22], where the fresh product itself has mainly been studied with respect to the DT. However, the application of the concept to a pharmaceutical air freight cool chain in the literature has been found to be limited. Therefore, the DT concept has been chosen in order to develop the real-time decision-making methodology for the improvement of operational quality. In specific, a DT development methodology introduced by Ait-Alla et al. [23] has been used as the basis for the development of the improvement method. This method has been used in order to construct a digital system containing a process virtualisation, or Physical Twin (PT), and a DT decision support module containing improved real-time decision-making. Given the research scope, the cool chain operational quality improvement methodology has been applied to a specific pharmaceutical air freight cool chain system through a case study at KLM Cargo. The resulting improvements have been quantified by means of a proposed novel effectiveness metric. Consequently, the contribution of this research project has been recognised as an extension of the application of the DT concept into the domain of the pharmaceutical air freight cool chain in order to improve operational quality by means of real-time decision-making.

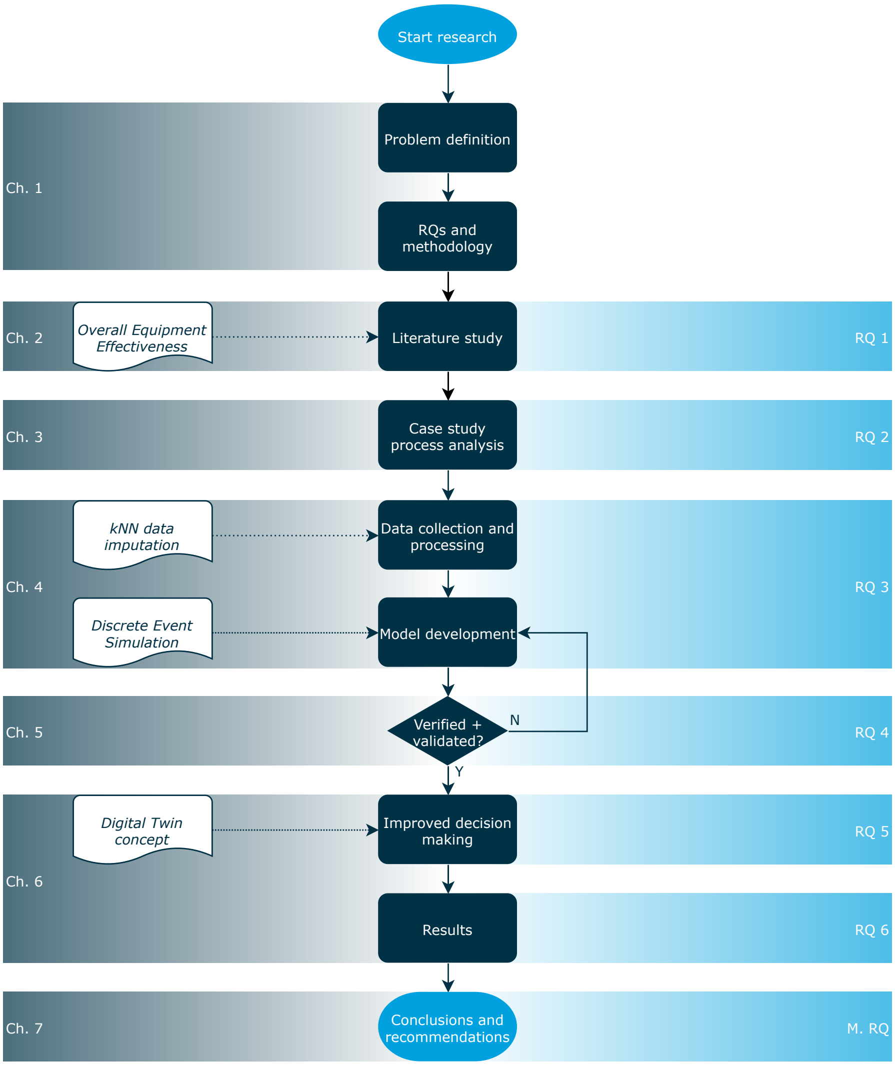


Figure 1.1: Conceptual model of the research project

2

Current state of the art of cool chain management

In the following chapter, the relevant and required literature and methodologies for the development of a cool chain management improvement methodology have been presented. Firstly, a general overview of the air freight industry has been provided in Section 2.1. Thereafter, a literature overview on cool chain management has been provided in Section 2.2. Then, the relevant literature on missing data in industry has been discussed in Section 2.3. Consequently, the evaluation of a system such as a cool chain has been discussed in Section 2.4. In Section 2.5, relevant modelling techniques have been discussed for the implementation and execution of the research. Then, the suggested concept for improved cool chain management has been elaborated on in Section 2.6. Finally, the company related to the case study has been introduced in Section 2.7. The chapter has been concluded with a synthesis of the previously mentioned sections and the relevant findings in Section 2.8, with the aim of answering the following research question:

1. Considering the state of the art, how can cool chain management be improved?

2.1. The air freight industry

In order to provide the required background information for a proper understanding of the air freight cool chain, a general discussion on the air freight industry has been provided. The air freight supply chain consists of three major actors: shippers, forwarders and carriers [4]. The shipper can be considered as the party who wants to ship cargo from one place to another. Consequently, the forwarder arranges the door-to-door transport of the shipment, handles the necessary documentation and possibly consolidates freight with the same destination into a single shipment. Finally, the carrier performs the airport-to-airport shipment. Kupfer et al. [24] have extended the air freight business model with the notion that an important distinction should always be drawn between integrated and non-integrated air cargo carriers in any discussion of air freight operations. The reason has been interpreted as the influence of the type of carrier on the involved actors and processes in the supply chain. The extended business model can be seen in Figure 2.1.

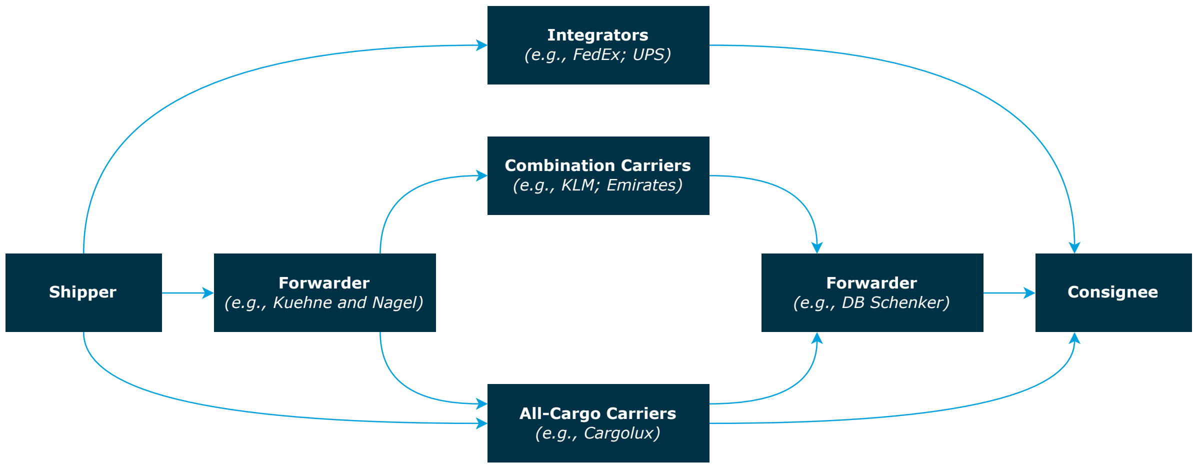


Figure 2.1: Air freight delivery business model, adapted from Kupfer et al. [24] and Debbage and Debbage [1]

Most companies in the air freight industry operate as non-integrated service providers, including forwarders, combination carriers such as KLM and all-cargo carriers such as Cargolux [24]. As the name suggests, combination carriers combine the capacity of aircraft for both passengers and their luggage and freight, while all-cargo carriers utilise aircraft solely for the transportation of freight and are thus not involved in the passenger business. However, the distinction between combination and all-cargo carriers is not always as clear as just indicated since many large passenger carriers also operate dedicated all-cargo aircraft besides the passenger fleets. Integrated carriers, or integrators, provide a full door-to-door solution using a combined fleet of aircraft and road vehicles, such as FedEx. Integrators tend to own all the assets of production throughout the entire logistics value chain [1] and are among the largest cargo airlines in the world. Although the approach of a combination carrier, all-cargo carrier or integrator differs, the basic service offered is essentially equal for all; the air transportation of freight. Considering a combination carrier such as KLM, the forwarder is the primary party which places bookings for freight transportation and may thus be considered as the customer. By zooming in and reducing the perspective to an airport or hub, additional actors in the supply chain become apparent, which has been shown in Figure 2.2.

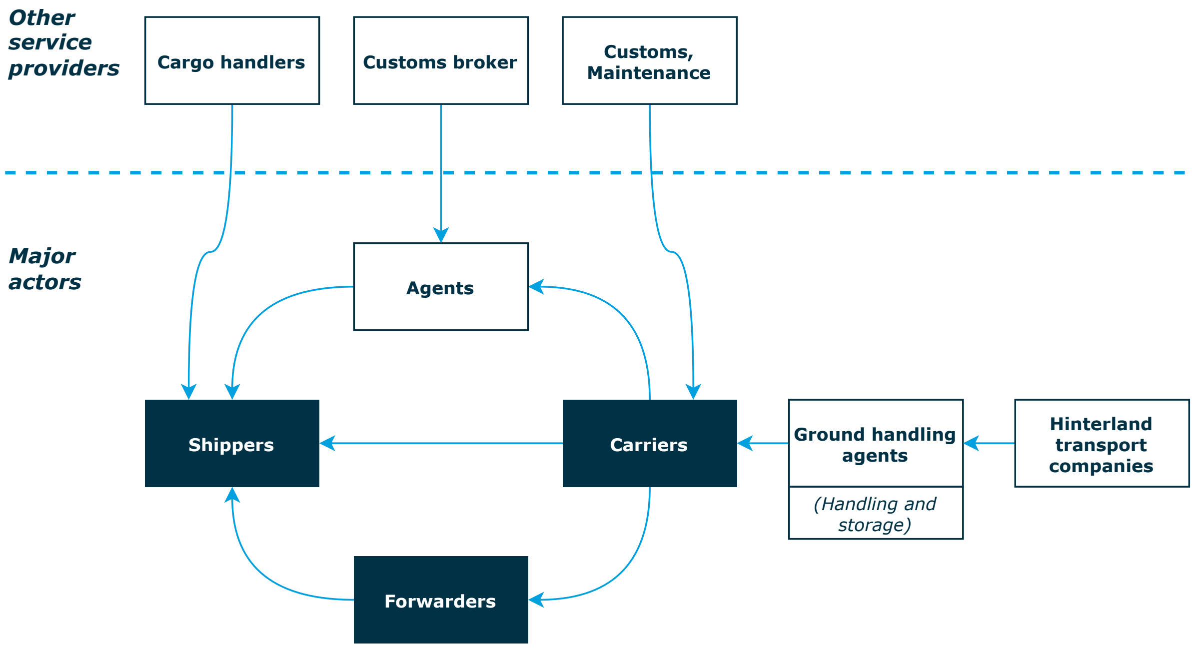


Figure 2.2: Air freight actors at an airport or hub, adapted from Merkert, Van de Voorde, and Wit [25]

In principle, each actor involved in the air freight supply chain constitutes an integral part of air logistics and provides services to other actors and ultimately the shipper. Additional major actors worth mentioning include the hinterland transport companies and terminal operating companies. Merkert, Van de Voorde, and Wit [25] have mentioned the phenomenon of frequent feeding of freight towards large intercontinental hubs such as Schiphol (SPL), which is mostly done through trucking in Europe. The actors which provide the feeding of freight towards the hub are the hinterland transport companies. Upon the arrival of freight at the hub, a terminal operating company, or Ground Handling Agent (GHA), arranges the handling, temporary storage and preparation of freight for air transportation.

2.2. Cool chain management

A cool chain is a system used for keeping and distributing environmentally sensitive products in the required conditions. Generally speaking, the system consists of a series of storage and transport links which have been arranged in such a way as to maintain pharmaceutical freight at the correct temperature. Unfortunately, in reality, a cool chain system is not ideal and may contain certain breaks, such as handovers in an intermodal connection. As an example, considering the air freight industry, pharmaceutical freight arriving by truck at an airport must be unloaded, handled and finally transported towards the aircraft. During such activities, it is not uncommon for sensitive freight to be exposed to harmful environmental conditions. In principle, loss of quality throughout a cool chain transportation network is a cumulative process in which each break adds up [26]. However, it is especially difficult to eliminate cool chain breaks in the air freight industry due to for instance the required warehouse and air-side handling at airports. Through experience, pharmaceutical companies are aware of the risks encountered throughout a cool chain. Therefore, appropriate packaging solutions are chosen in order to mitigate the risks of cool chain breaks. Nonetheless, the packaging has usually been optimised for specific external temperature ranges. Consequently, it is the responsibility of the shipper to define the acceptable temperature range for a shipment. The temperature range or limit thus only reflects the external or ambient handling temperature allowed during transportation and not the actual internal product temperature IATA [26]. Since cool chain management aims at preserving the quality of products throughout the chain [6], businesses in the pharmaceutical and medical industries are increasingly relying on cool chains [27]. The demand increase can largely be attributed to the growth of the biopharmaceutical industry [10]. Considering for example vaccines, Kartoglu and Milstien [12] have noted that the inherent sensitivity and instability differs between the types of vaccines. For instance, there are two original types of vaccines; live viral and bacterial or inactivated vaccines. Live vaccines are typically freeze sensitive and also not stable to high temperatures, whereas inactive vaccines cannot be frozen yet are more stable to heat. In principle, the cool chain was developed for these two types of vaccines. However, especially with the growth of the biopharmaceutical sector, many newer vaccines and other pharmaceutical products cannot be easily divided into the two categories. Furthermore, such vaccines show a wider range of behaviour in terms of heat stability and freeze sensitivity. Therefore, the necessity for careful attention at all levels of the cool chain has become apparent. Besides the fact that cool chain problems occur globally as noted by Kartoglu and Milstien [12], it has been mentioned that the cool chain is typically thought of as a protection mechanism from excessive heat. However, it is imperative to note that low temperatures are an equally important hazard for the integrity of pharmaceutical shipments [12, 28, 29] and should thus also be considered in cool chain management. The fact that pharmaceutical products may be more sensitive to either excessive cold, heat or both, accentuates the importance of maintaining shipment temperature within the specified temperature range. Despite efforts such as product packaging, cool chain failures remain present during day-to-day operations. However, for effective cool chain management and the improvement thereof, it is imperative to consider the types of failures that may occur in such a system.

Since a cool chain has been regarded as a supply chain with the additional aim of preserving product quality [6], cool chain management and the related challenges may be expected to parallel supply chain management. Indeed, considering a three-fold understanding of a supply chain in terms of physical movement of goods, informational flows and decision making, Comes, Sandvik, and Van de Walle [30] have attributed cool chain failures to a failure in any of these categories:

- The lack or failures of the physical layer leading to a disruption of material flows;
- Information gaps and the lack of the ability to manage flawed information;

- A failure of decision-making, coordination or planning.

Furthermore, from a humanitarian supply chain perspective, the categories have been related to specific cool chain challenges in Table 2.1 while a cool chain analysis framework has been proposed. An adaptation of the framework can be seen in Figure 2.3.

Category	Disruptions
Disruption of material flows	Critical infrastructure failure, e.g. power blackout, disruptions of transportation network, closed warehouses [31]; Failure of equipment and lack of redundancies, e.g. lack of fuel, spare parts and back-up energy [32]
Information gaps	Failure of monitoring and tracking systems; incorrect use of vaccine vial monitors; no tracking of minimum and maximum temperatures [29]; Breakdown of communication and information systems; Lack of ability to manage the complex information stream, and work with delayed, lacking or uncertain information
Failure of decision making	Deficiencies in vaccine storage and handling and lack of training [33, 34]; Lack of mitigation and management options for possible disruptions and lack of planning [35]; Lack of operational decision support [36]

Table 2.1: Cool chain disruption taxonomy [30]

As shown in Table 2.1, a wide range of cool chain disruptions and challenges can be encountered, which generally can be associated with the three failure categories. A disruption of material flow is typically attributed to a lack of infrastructure or equipment, or the failure thereof. Furthermore, a lack of the ability to extract useful information from the cool chain may lead to information gaps, which increase the difficulty to assess the performance of the system. Finally, related to information gaps, there may in general be a failure of decision making or lack of operational support available. This could be attributed to both a lack of information available or the inability to utilise available data for improved decision making. Figure 2.3 depicts the different layers and their dependencies as related to the cool chain disruption taxonomy which has been shown in Table 2.1.

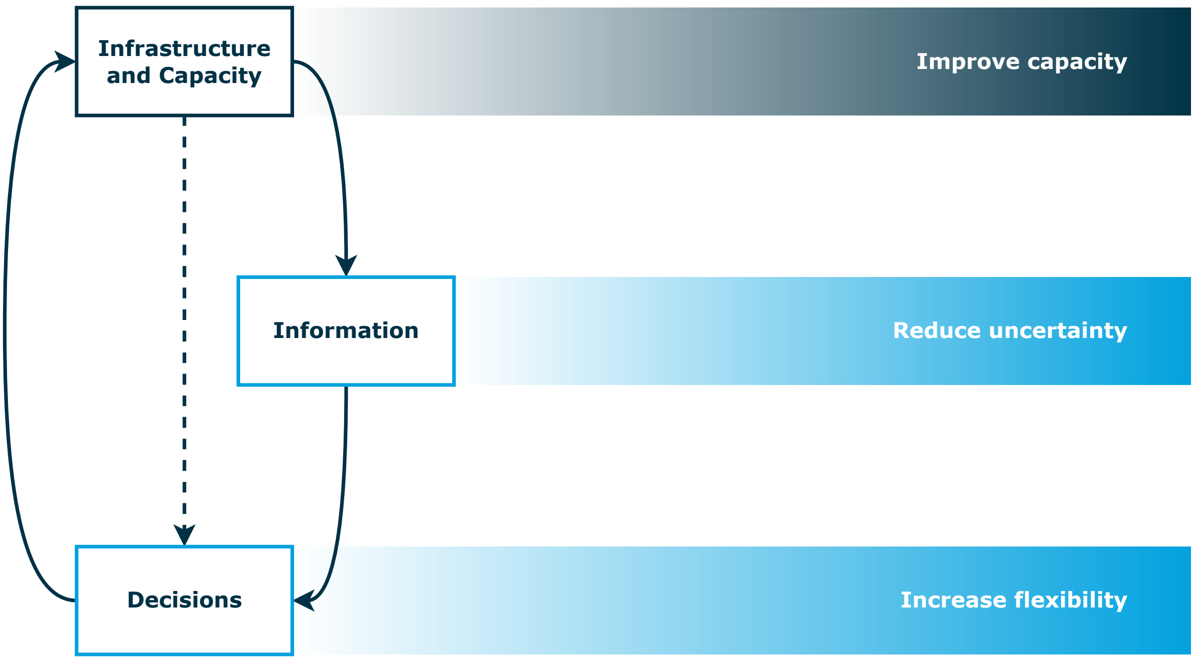


Figure 2.3: Analysis grid for cool chain review, adapted from Comes, Sandvik, and Van de Walle [30]

The infrastructure and capacity layer enables and generates the information and communication flows. Such information flows then enable and support the decision making which in turn impact the physical layer through for instance possible interventions or resource allocation. Furthermore, the framework shown in Figure 2.3 indicates that for effective decision making, supporting information is to be extracted from the physical layer by active monitoring of the process. Considering Table 2.1, Comes, Sandvik, and Van de Walle [30] have mentioned that technologies are increasingly used to overcome the specific disruption categories in the different layers by; improving capacity in order to address infrastructure problems, reducing uncertainties or information gaps and by increasing flexibility through improved decision making. However, improving capacity is generally not a viable option for any short term improvements since apart from significant investments, the lead time and procurement of cool chain equipment may take up to two years [13]. Nonetheless, the unavailability of cool chain facilities has been recognised as a typically encountered problem in practice, which is simply caused for instance by cool storage being full [37, 38]. Therefore, information extraction and improved decision making may provide a suitable method for cool chain management improvement. In fact, regarding the information and decisions layers, Han et al. [39] have noted that for a cool chain, one of the main research directions is to ensure the integrity of the cool chain and its precise control. Given the research goal and scope, this work has been primarily focused on the information and decisions layers portrayed in Figure 2.3.

Considering the literature with regards to cool chains in general, significant efforts have been spent on the improvement of such systems, especially in the information and decisions domain shown in Figure 2.3. As an example, Wang, Kwok, and Ip [40] have developed a real-time monitoring and online decision support system with Radio Frequency Identification (RFID) and a sensor network and a decision rule base for the improvement of perishables transportation. The results from the simulation have shown that the monitoring and decision support system is an efficient tool for reducing the transportation losses of perishable products for the enterprises in cold chain. Askin, Khodadadegan, and Haghnevis [41] have developed mathematical decision models that consider remaining shelf life in determining the dynamic assignment of perishable items in warehouses, using environmental data collected with RFID technology. Similarly, a significant amount of research has been performed on shelf life prediction and product quality status assessment throughout fresh cool chains [42, 43, 44, 45]. However, given the wide range of pharmaceutical product characteristics and limited information sharing between cool chain partners, methods for improvement in the food cool chain are not easily transferred to the pharmaceutical cool chain. As previously mentioned, pharmaceutical companies choose appropriate packaging. However, since technical details on the packaging such as thermal conductivity and the product itself is not shared, it is not possible to accurately determine shelf life as seen in the literature for fresh cool chains. Consequently, pharmaceutical cool chain improvement in the air freight industry, has received less attention in the literature especially with regards to digitisation. Higgins et al. [46] have proposed a configuration designed to improve data management for real-time analysis of sensor data, for which a pilot has been performed in a pharmaceutical cool chain containing air shipping lanes. Terpstra, Zhang, and Akçay [47] have attempted to utilise available data in a pharmaceutical air freight cool chain in order to predict temperature profiles of new shipments. However, Ashok, Brison, and LeTallec [13] have noted that an understanding of cool chain performance is typically limited due to a lack of performance management systems. Furthermore, a brief understanding of cool chain performance may be available from infrequent assessments, yet there are rarely routine systems in place to provide consistent insight into cold chain performance and enable day-to-day performance management. At the same time, Kartoglu and Milstien [12] have noted that the regulatory trend has been aimed towards increased oversight, management and control of environmental conditions across the entire supply chain. In general, supply chain risk managers and especially cool chain managers, are interested in decision-making support in order to monitor and recognise disruptions in real time while being able to determine the required actions to deal with such situations [48, 49, 50, 51]. Some researches have pointed out a trend in supply chain management towards a Digital Twin (DT), i.e. computerised models which represent a physical object in real time [52, 53, 54, 55, 56]. In fact, Haße et al. [57] have mentioned that a DT can offer considerable potential, especially in logistics. A DT which represents a physical supply chain based on actual system data has the potential to be used for planning and real-time control decisions [58]. Similarly, Verdouw et al. [59] have addressed that object virtualisation within supply chains has been an important topic in research. Furthermore, the authors have mentioned that virtualisation allows for the decoupling of physical flows and information aspects.

Consequently, virtual objects such as freight can be enriched with sensor data about properties such as temperature information, which allows for advanced control capabilities including tracking and tracing, quality monitoring and planning functionalities. Moreover, in order to advance a transport system such as the cool chain towards an intelligent system, it is necessary to monitor the system and collect information, model and predict dynamics and finally control the process for optimal performance [60]. Therefore the concept of a DT has been deemed as an appropriate method for the improvement of the air freight pharmaceutical cool chain. Nonetheless, the quality of model-based decision-making support in supply chains crucially depends on the availability of data and the quality thereof [58]. Accordingly, in Section 2.3 an overview of methods for handling missing data has been provided. Consequently, a novel method for assessing cool chain performance has been investigated in Section 2.4.

2.3. Missing data imputation

Throughout many industries, an unfortunate yet frequently occurring issue is that of missing data. Besides operational issues encountered with missing data, any absence of data may cause bias in statistical analysis while making many data modelling techniques ineffective [61, 62]. Consequently, effective methods for handling missing data have become a necessity. Such common known methods range from data omission to sophisticated imputation algorithms [63]. Data omission is commonly applied and simply entails the removal of data samples which contain missing data values. Although it is an inherently simple method, it may significantly decrease the sample size of a data set. Furthermore, any deletion of samples may result in discontinuous time-series data [64]. On the other hand, the goal of data imputation is to generate plausible replacement values of missing data such that sample omission is not necessary [65]. However, before considering available data imputation methods in the literature, it has been deemed appropriate to further elaborate on the characteristics of missing data. In specific, missing data patterns and missing data mechanisms have been further discussed in the following subsection, followed by the different imputation methods.

2.3.1. Missing data patterns and mechanisms

Ehrlinger et al. [66] have noted the fact that no standardised list of missing data patterns exists in the literature. Therefore, the authors have described the three most occurring patterns as well as two additional patterns of special interest for the industrial domain [66]. The respective patterns have been visualised in Figure 2.4.

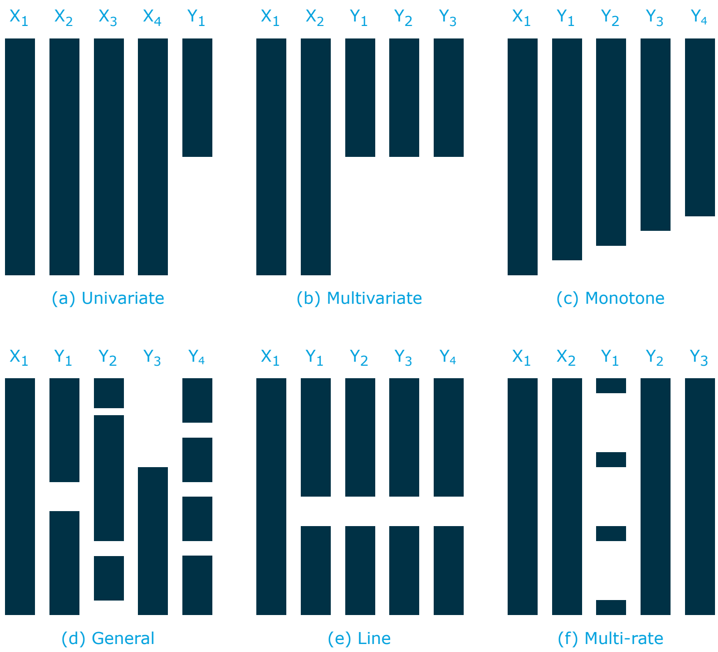


Figure 2.4: Missing data patterns, adapted from Ehrlinger et al. [66]

Each missing data pattern has been further elaborated on following the work by Ehrlinger et al. [66]:

1. **Univariate and multivariate pattern**: the univariate pattern in Figure 2.4 *(a)* has been considered as the simplest case, where exactly one variable Y_1 contains missing values. The multivariate pattern in Figure 2.4 *(b)* has been considered as a special case of the univariate pattern, containing a set of variables with missing data, in this case, $Y_1, \ldots, Y_3$. Univariate and multivariate patterns may occur with the failure of one or more sensors.
2. **Monotone pattern**: the monotone pattern shown in Figure 2.4 *(c)* is typical for social sciences where participants tend to leave a study which is conducted over time. However, the monotone pattern has little significance in the industrial domain.
3. **General pattern**: the general pattern shown in Figure 2.4 *(d)* is the default missing data pattern encountered in practice. Despite the frequency of occurrence of this pattern, it has also been considered the most difficult pattern to handle, since it is often a combination of other patterns. The causes may be the co-occurrence of sensor failures and manual data removal.
4. **Patterns for industrial applications**: the line and multi-rate patterns shown in Figure 2.4 *(e)* and *(f)* respectively have been denoted as of specific interest for the industrial domain [67]. The line pattern is typically attributed to sensor breakdown. In contrast, the multi-rate pattern does not necessarily represent an error since this pattern typically occurs intentionally. As an example, specific features may be measured less often than other features in the data set.

Besides the pattern of occurrence of missing data, several mechanisms of missing data have been recognised which describe the relationship between missing and existing values [68]. The three mechanisms have been further discussed following the work by Ehrlinger et al. [66] and Zhang and Thorburn [64] and have been visualised in Figure 2.5.

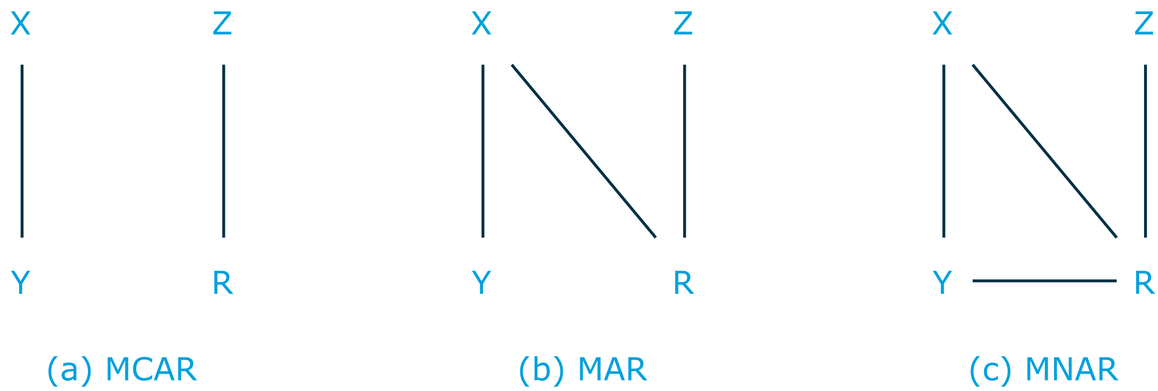


Figure 2.5: Missing data mechanisms, adapted from [69]

Figure 2.5 indicates how the missingness R is affected by the complete variables X, the partially missing variables Y and external causes Z. The three mechanisms have been further discussed in relation to Figure 2.5:

1. Missing Completely at Random (MCAR): if the missing data is completely at random, the missingness only depends on external causes Z, which have no influence on the overall data, as seen in Figure 2.5 *(a)*. In this sense, MCAR is an ideal situation with regards to imputation, yet does not occur frequently in practice.
2. Missing at Random (MAR): when data is missing at random, missing values are not only caused by Z but are also influenced by complete variables X, as shown in Figure 2.5 *(b)*. It is therefore possible to distinguish MAR from MCAR by identifying patterns in X which describe the missingness. For example, if all sensor readings for a specific machine are missing, the data is MAR.
3. Missing Not at Random (MNAR): with respect to Figure 2.5 *(c)*, data which is missing not at random is influenced by three factors: Z, X and the observed variable which contains missing

values Y. An exemplary cause of MNAR could be the missing of values for sensor readings above a certain temperature. In principle, in this case, the missing data is systematically related to the unobserved data and is thus significantly difficult to handle.

In general, since any information about the missing values is not available, it is difficult to automatically distinguish between MAR and MCAR. In such cases, it is only possible to collect additional information on the missing data.

2.3.2. Imputation methods

Zhang and Thorburn [64] have classified the imputation methods according to three groups: statistical-based methods, model-based methods and neural network-based methods. Each group has been briefly discussed below.

Statistical based

In statistical-based imputation methods, missing data values are replaced with plausible values which can be derived by substituting values from the available observed variables [70]. Three commonly used statistical imputation methods are *mean imputation*, *last observation carried forward* and *linear imputation*. As the name suggests, mean imputation involves the replacement of missing values with the arithmetic mean of the other available values. With the last observation carried forward method, missing values are imputed from the last observation in the data set. It is evident that with this method, a rather unrealistic assumption is made that there has been no change at all since the last measured observation [71]. Finally, with linear imputation, missing values are based on adjacent available values through linear interpolation. Linear imputation is a preferred method for estimating continuously missing data over a short time interval. This is due to the fact that the accuracy of linear imputation typically decreases as the length of the missing data period increases [64].

Model based

Model-based imputation has the goal of building predictive models in order to impute estimated values for each target variable which contains missing values. Several commonly deployed methods include: *Expectation-Maximisation (EM)*, *Multiple Imputations by Chained Equations (MICE)* and *k-Nearest Neighbours (kNN)*. EM is a parametric method in which missing values are imputed based on the maximum likelihood estimation. The method contains two steps in which firstly missing data is estimated based on all observed data and estimation model parameters, after which the expectation of the full data set is maximised to obtain the next guess of missing data. Both steps are iterated until the model converges and the missing data can be estimated. In the MICE procedure, a series of regression models are run whereby each variable with missing data is modelled conditional upon the other variables in the data [72]. Finally, kNN is a popular approach in data processing applications. In principle, it has been designed to replace missing values by using k-most similar non-missing data. The kNN algorithm is therefore used to search the entire data set for the k number of most similar cases, or neighbours, which show the same patterns as the row with missing data [73]. The missing data value is then computed based on the mean values of the kNN values.

Neural network based

Given recent advancements in the application of Deep Neural networks in various fields, such methods have likewise been applied in estimating missing data. Two main architectures for neural network-based imputation models have been considered: *sequence-to-sequence model* and *recurrent neural network*. In sequence-to-sequence-based models, missing values are imputed directly based on predictions whereas for recurrent neural networks missing data is estimated when computing other correlated prediction tasks. The exact inner workings of either model architecture have been deemed beyond the scope of this work and have thus not been discussed any further.

Given the inherent differences between the discussed imputation groups, several limitations have been recognised by Zhang and Thorburn [64]. Regarding statistical-based methods, missing values are replaced by values as prescribed by a certain rule. Although these methods are computationally simple, the relationships between variables in the data set are ignored. Therefore, statistical methods tend to lead to an underestimation of the variability in the data set [74]. Furthermore, statistical-based methods assume all missing data follows a constant pattern. For example, missing values are close to the mean value with mean imputation or the preceding available value with the last observation carried forward technique. As a consequence, statistical-based methods are often potentially biased and are to be used with significant caution [75]. Regarding model-based methods, the relationship between different variables is taken into account through the creation of regression models for missing values which take the non-missing values as inputs [76]. However, the creation of such models may lead to significant computational overhead. Finally, neural network-based methods can offer advantages that are not offered through different methods, such as the ability to capture long-term temporal dependencies. However, neural network models are typically non-transparent [77], require significant time to train [78] and are highly dependent on hyper-parameter tuning [79].

2.4. System evaluation

In order to assess the operational quality, as defined in Chapter 1, of a system such as a cool chain, certain performance indicators are required. The Overall Equipment Effectiveness (OEE) indicator has been introduced by Nakajima [80] within the Total Productive Maintenance conceptual framework. In principle, OEE is a metric which can be used to measure the effectiveness of production equipment and how effectively a manufacturing operation is realised [81]. The original definition of OEE has been expressed as a percentage resulting from the multiplication of three measures, which can be seen in Equation 2.1.

$$\text{OEE} = \text{Availability rate} \cdot \text{Performance rate} \cdot \text{Quality rate} \tag{2.1}$$

where each measure has been defined as:

- **Availability**: the actual time used versus the planned time;
- **Performance**: the actual production versus the standard during the actual time used in production;
- **Quality**: the number of faulty products produced in comparison to the total number produced.

Although the OEE is computed in percentages, time is the central metric unit for the respective submeasures [82]. As a main advantage of using the OEE, the most important factors influencing equipment performance are allowed to be monitored while clearly identifying root causes for losses in manufacturing effectiveness [83, 84]. In the literature, there has been some confusion on whether the OEE indicator refers to efficiency or effectiveness as depicted in the name [85]. Muchiri and Pintelon [86] have noted that effectiveness is defined as a process characteristic that indicates the degree to which the process output conforms to the requirements while efficiency is defined as a process characteristic indicating the degree to which the process produces the required output at minimum resource cost. Furthermore, Muchiri and Pintelon [86] have indicated that the three measures captured by the OEE indicate the degree of conformation to output requirements. Therefore, the conclusion has been drawn that the OEE is indeed a measure of effectiveness, which in other words thus measures the degree to which equipment is doing what it is supposed to do. Nonetheless, it should be noted that the OEE may measure effectiveness, efficiency or both depending on the formulation of the metric. In terms of application, OEE has been widely used to control production systems and verify operational improvements [87]. Besides equipment in a production environment, the OEE methodology for measuring the effectiveness and or efficiency of a system has been applied to other fields. As an example, Pinto, Goldberg, and Cardoso [88] have applied the OEE indicator to benchmark the operational efficiency of port terminals. Muñoz-Villamizar et al. [89] have applied OEE to evaluate the effectiveness of urban freight transportation systems. Similarly, García-Arca, Prado-Prado, and Fernández-González [90] have adapted the OEE for usage in road transportation management for the improvement of the overall system efficiency. Therefore, it can be concluded that the concept is generally acceptable while its use can be extended beyond the application in manufacturing [91]. Consequently, it has been deemed possible to adapt the OEE to match the research object; the cool chain. Furthermore, an adaptation of the OEE methodology to fit the need for cool chain management may introduce a novel performance assessment and management tool. Interestingly, Kaiblinger and Woschank [92] have found that, among other concepts, OEE was most often addressed as the main objective in reviewed DT studies. Although no paper has mentioned OEE specifically, many studies have considered this performance indicator through the consideration of the underlying indicators; quality rate, machine availability and machine performance. Therefore, this work may provide additional novelty through the explicit application of an OEE based performance indicator in a DT development study.

Based on the OEE performance measurement methodology, a new framework has been constructed to suit the characteristics of a cool chain: the Overall Cool Chain Effectiveness (OCCE). In comparison to the traditional OEE, which is typically meant for one equipment [91], the OCCE has been proposed as a measure for the effectiveness of a cool chain as a whole, or part of it depending on the research scope. Characteristically for a cool chain in general is the duality in operational quality with regards to the timeliness of freight handling as well as the conformity to required environmental conditions. In essence, the metric indicates to which degree the cool chain is doing what it is supposed to do and thus quantifies the conformation to output requirements of the system. A formal definition of the proposed OCCE metric has been shown in Equation 2.2.

$$\text{OCCE} = \text{Cool Storage Availability} \cdot \text{On Time Performance} \cdot \text{Temperature Adherence} \tag{2.2}$$

Similar to any type of equipment, the effectiveness of a cool chain may also be represented by three rates as shown in Equation 2.2. Each respective rate has been further discussed:

- **Cool storage availability**: cool storage availability refers to the utilisation of cool storage facilities. As mentioned in Section 2.2, the unavailability of cool storage facilities is a typical problem encountered in cool chains. Therefore, the availability rate indicates to which extent a cool chain provides the availability of required infrastructure.
- **On time performance**: in principle, the quality of a cool chain can be seen as twofold; on the one hand timeliness and on the other the extent to which freight is handled according to the required environmental conditions. The on time performance provides an indication of to which extent the timeliness of a cool chain is as per request.
- **Temperature adherence**: finally, the temperature adherence provides an indication of to which extent freight is handled according to specification by the shipper.

Combined, the OCCE measure thus provides an overview to which extent a cool chain conforms to the requirements, based on the availability of critical infrastructure, the timeliness of freight processing and finally the adherence to temperature restrictions for the sensitive cargo.

2.5. Modelling techniques

In general, Law, Kelton, and Kelton [93] have elaborated on the multiple ways to study a system, which can be seen in Figure 2.6.

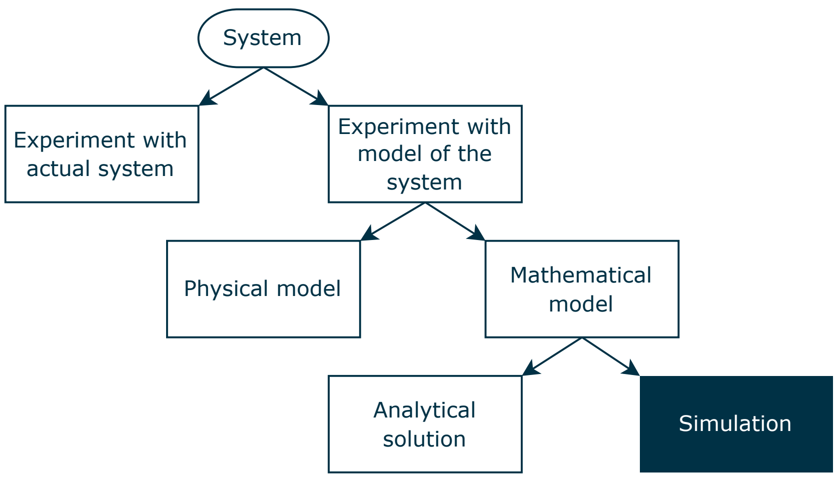


Figure 2.6: Methods for studying a system, adapted from Law, Kelton, and Kelton [93].

In principle, depending on the research object, it is possible to experiment with the actual system. However, more commonly it is either too disruptive or too costly to do so, therefore necessitating the need for a model of the system under study. A model may then be classified as either physical or mathematical in nature. A physical model is typically not of interest in systems analysis whereas the vast majority of models built for such purpose are mathematical [93]. A mathematical model represents a system in terms of logical and quantitative relationships. Finally, a mathematical model may provide either an analytical solution or a simulation. Since many systems are significantly complex, analytical solutions are usually not available. Furthermore, simulation models are among the most widely used quantitative approaches in the modelling of production and logistics systems, in turn allowing to simulate the operation and decide on any operational aspects [94, 95]. Therefore, given the complexity and specificity of the research object, a cool chain can be adequately studied through simulation modelling.

Simulation models have been classified along three different dimensions by Law, Kelton, and Kelton [93]:

- **Static versus dynamic simulation models**: as the name suggests, static simulation models represent a system at a specific time or a system in which time plays no role. On the contrary, a dynamic simulation model represents a system that evolves over time.
- **Deterministic versus stochastic simulation models**: a simulation model which does not contain any probabilistic components can be classified as a deterministic model. However, many systems are modelled with random or probabilistic components, giving rise to stochastic simulation models.
- **Continuous versus discrete simulation models**: a continuous system is characterised by state variables which change continuously as time evolves. On the other hand, discrete systems have state variables which change instantaneously at separate points in time. It has been noted that a discrete model is not always used to model a discrete system.

Given the described simulation model dimensions, any simulation model for a cool chain system may be described as a dynamic, stochastic and discrete model. Undoubtedly, a cool chain can be considered as a dynamic system, since freight progresses throughout the chain and environmental conditions evolve as time passes. Furthermore, given the complexity of a cool chain, deterministic modelling is rarely possible for such a system and most queuing and inventory systems are modelled stochastically [93]. Finally, depending on the research objective, a cool chain can be either modelled in a discrete or continuous manner. For example, a study focusing on intrinsic fresh or pharma product quality degradation may adopt a continuous simulation modelling approach. However, from a logistics perspective, the movements of and interactions with freight can be considered as a discrete system. Although temperature does evolve naturally and continuously, temperature sensors discretise temperature measurements which are then usually logged at fixed intervals. Therefore, a cool chain has been considered a discrete system. Two frequently applied simulation modelling techniques have been further discussed in the following sections.

2.5.1. Agent-Based Modelling and Simulation

In recent years, Multi-Agent Systems (MAS) have received increased attention in the literature. MAS consist of autonomous entities known as agents [96]. Agents are able to collaboratively solve tasks through an inherent ability to learn and make autonomous decisions. This ability arises from the fact that agents use the interactions with neighbouring agents or with the environment to learn new contexts and actions [96]. By using its respective knowledge, each agent is able to decide and perform an action on the environment to solve a task. It is this flexibility that makes MAS suitable for solving problems in a variety of disciplines including computer science and electrical engineering [97].

The concept of MAS has been applied to the basic structure of simulation models in order to obtain Agent-Based Modelling and Simulation (ABMS) [98]. In ABMS, active components or decision makers are conceptualised as agents, which are consequently modelled and implemented using agent concepts. Such concepts include for instance explicit goals which drive agent behaviour, the ability to learn and adapt and the environment in which it is situated [99]. The key idea of ABMS hinges on the notion that a global phenomenon can be generated from the actions and interactions of the MAS [98]. Therefore, Klügl and Bazzan [98] have mentioned that ABMS has been found especially suitable for the analysis of complex adaptive systems and emergent phenomena in social sciences, traffic, biology and others. Furthermore, it has been mentioned that ABMS is able to provide an improved understanding of real-world systems in which the representation or modelling of many individuals is important and for which the individuals have autonomous behaviours [100].

2.5.2. Discrete Event Simulation

Similar to the description of a discrete system, Discrete Event Simulation (DES) concerns the modelling of a system as it evolves over time through a representation in which the state variables change instantaneously at distinct points in time [93]. The discrete points in time are defined by the moments an event occurs, where an event has been defined as an instantaneous occurrence which may change the state of the system. Given the dynamic yet discrete nature of the modelling approach, a suitable time advancement mechanism is necessary. Time in a DES, which could be in any defined unit such as hours or minutes, is recorded and represented by the simulation clock. In order to advance the simulation clock, a next-event time advance approach has usually been applied. In essence, the simulation clock is advanced to the discrete point in time at which the next event occurs, based on the simulation event list or calendar. Generally, a DES is run until all events on the event list have been handled or a specific stopping condition has been satisfied. Each DES model contains an executive routine for the management of the event calendar and simulation clock, which ultimately arranges the sequencing of events and drives the simulation [101]. Once the next event becomes scheduled and consecutively active, control is transferred to the appropriate routine. Such an operation routine may be an event, activity or process and depends on a so-called world view [102]. Three world views have been distinguished for DES modelling [101]:

- **Event scheduling**: each type of event has a corresponding event routine, which is scheduled on the event calendar by the executive routine.
- **Activity scanning**: a simulation contains a list of activities, each defined by two events; the start event and the completion event. Moreover, each activity contains test conditions and actions. The executive routine scans the activities for satisfied time and test conditions and executes the actions of the first selectable activity.
- **Process interaction**: the worldview focuses on the flow of entities through a model. The process interaction strategy views systems as sets of concurrent, interacting processes. Furthermore, a process class describes the behaviour of each class of entities during its lifetime throughout the simulation, for example, freight entities flowing through a cool chain process. The executive routine uses a calendar to keep track of forthcoming tasks. Additionally, the state in which the process was last suspended is recorded.

Since a logistics system such as the cool chain contains flowing entities, i.e. freight, the process interaction simulation strategy is highly suitable and has thus been chosen. An additional advantage is the fact that a process interaction program structure maintains a closer relation to the model structure and consequently a modelled cool chain.

Regarding production and logistics systems, DES is in fact the most used simulation technique [103]. Similarly, Kaiblinger and Woschank [92] have noted that for production logistics processes, DES is most often used since it is also the state-of-the-art for simulating production logistics systems. Furthermore, Siebers et al. [100] have noted that DES is useful for problems which consist of queuing systems or complex networks of queues where many of the applications occur in manufacturing and service industries. Since a cool chain can be seen as a complex logistical queuing network, DES has been deemed as an appropriate simulation modelling technique. Moreover, Kaiblinger and Woschank [92] have found that, in the context of production logistics, DES is most often used in order to create a virtual model for the development of a DT. With regards to the implementation of a DES, several options are available: manually programming a DES using a programming language such as Python or using a software package to develop a simulation model. In principle, software packages provide an intuitive way of developing models through user interfaces, ultimately with the potential of providing faster model development times. However, compared to manual programming, software packages have been found to be limited in terms of modelling flexibility. Therefore, in order to model a cool chain and implement the DT improvement concept, manual programming has been recognised as a preferred option. The DT concept has been further introduced in the following section.

2.6. The Digital Twin concept

The definition of a DT has been specified by Schluse and Rossmann [104] as a virtual representation of a real-world subject or a real-world object which contains models of its data, functionality and communication interfaces. Similarly, Nguyen et al. [105] have noted that a DT refers to the digital representation of non-living and living physical objects and have even stated that it has soon become one of the key technological enablers in the new era of the digital economy and society. In comparison to Schluse and Rossmann [104] however, Nguyen et al. [105] have noted an essential characteristic of a DT: the capabilities to generate virtual instances and control the changes of a physical object in *real-time*. A more complete definition of a DT adopted by Hofmann and Branding [106] is that of a virtual and computerised counterpart of a physical system, used to simulate it, exploiting real-time synchronisation of data [107]. Furthermore, Hofmann and Branding [106] have distinguished three stages of integration of a DT, depending on the degree of automation of the data flow between the physical and digital object, which has been visualised in Figure 2.7.

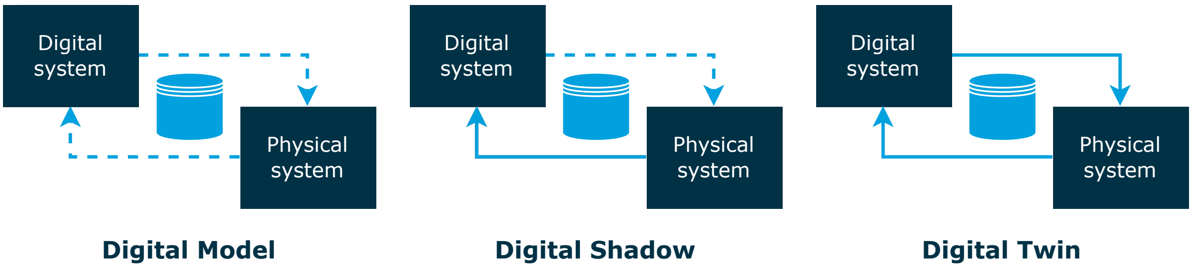


Figure 2.7: The three stages of Digital Twin integration depending on the automation of data flow, adapted from [107]

A digital model can be understood as a digital representation connected through a manual data flow, which infers that state changes do not have a direct impact on the physical system and vice versa. A digital shadow extends the concept of a digital model through an automated data flow from the physical to the digital system. Finally, a DT may then be considered as containing an automatic data flow in both directions. Coelho, Relvas, and Barbosa-Póvoa [108] have summarised the characteristics of a DT into six dimensions:

1. **The physical entity**: the studied physical existence which could be an activity process, device or product.
2. **The virtual system**: models intended to reproduce physical characteristics such as geometries, physics, rules and behaviours such that it is possible to replicate physical entities conveniently and reliably [109]. The virtual system consequently interacts with the physical system through control commands. Besides, it can provide improved policies to the service system module.
3. **The service system**: the system which integrates different service systems, such as a WMS.
4. **the data integration**: different types of data may be utilised in a DT system, Furthermore, with regard to integration, the origin of data is also variable. For example, data may be obtained directly from the physical system, whereas other data can be integrated from the virtual or service systems.
5. **The decision support system**: a decision support system has been noted as an important dimension of a DT. This is due to the fact that it enables the interaction of the decision-maker with the digital system as a whole. In principle, a decision support system receives information from the virtual simulation model and allows the decision-maker to interact with the system in real time.
6. **The connections between the systems**: finally, connections can also be considered highly relevant for a DT. Indeed, Tao and Zhang [109] have mentioned the fact that all dimensions

are interconnected and interact with each other in real-time enabling a DT to be consistent and optimised iteratively.

From the delineated six dimensions of a DT, it has become evident that a DT may constitute more than an exact digital representation of a system. In fact, Ait-Alla et al. [23] have stated the assumption that a DT should add additional functionality besides a virtual representation and have thus proposed a suitable methodology for DT development studies typically done by means of simulation, containing three different systems: the physical system, the Physical Twin (PT) and the DT. The methodology has been visualised in Figure 2.8.

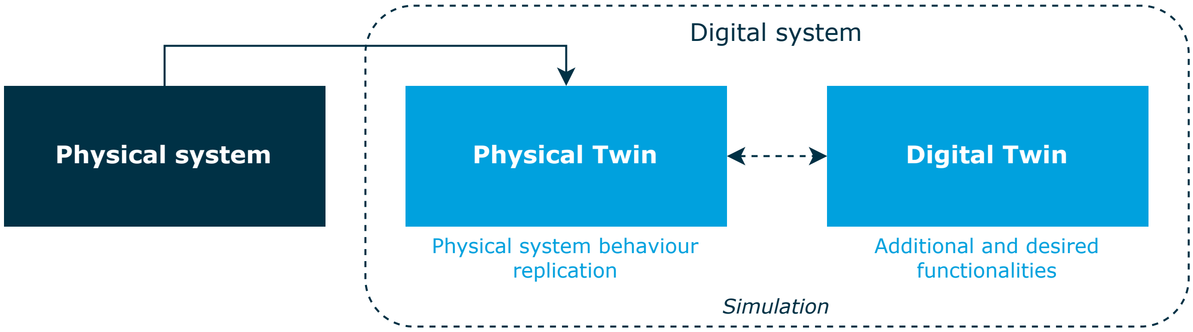


Figure 2.8: DT development study approach, adapted from Ait-Alla et al. [23]

As the name suggests, the physical system describes the system existing in the real world. Consequently, the Physical Twin constitutes a simulation model of the real-world system. In principle, the simulation model must not extend the functionalities of the real-world counterpart. Rather, the logic and physical properties of the physical system should be replicated to a sufficient degree for testing the indented functionalities of the DT and the PT can thus be considered as an intermediate model which separates the virtual representation from the actual DT model. Therefore, the Physical Twin could be considered analogous to the digital model referred to by Hofmann and Branding [106] and Kritzinger et al. [107]. Finally, Ait-Alla et al. [23] have described the DT as the digital representation of the physical system which, in contrast to the PT, adds additional and desired functionalities not covered by the physical system. Therefore, with regard to the six dimensions of a DT, several parallels have been found. Logically, the first dimension of the physical entity relates to the physical system as seen in Figure 2.8. Furthermore, the virtual system dimension has been recognised as the PT, while the actual DT can be considered as the additional functionality offered by a decision support system implemented into the digital system. The remaining dimensions have been found to be related to the respective interconnections between the physical system and the digital system.

Consequently, considering the current state of the art, a DT can be considered as a digital system which constitutes a virtual copy of a real physical system, including automatic data connections. However, from a development point of view, such a digital system may be built up from the PT part and the additional functionality offered by the DT. Consequently, a DT can then be defined as the additional functionality offered in the digital system, which interacts with the PT system representation. Following the presented definition, a DT may constitute different functionalities ranging from product quality prediction to decision-making support. Furthermore, the presented definition of a DT as an additional functionality which interacts with the PT in a digital system infers the fact that the DT cannot function without the PT model. Certainly, it has been argued that the intended and requested functionalities from a DT cannot be successfully utilised without a modelled virtual representation of the physical system: the PT. In conclusion, a DT has been defined as the additional functionalities offered by the digital system through interactions with the PT model representation of the physical system, while utilising automatic data connections. The definition of a DT has been visualised in Figure 2.9.

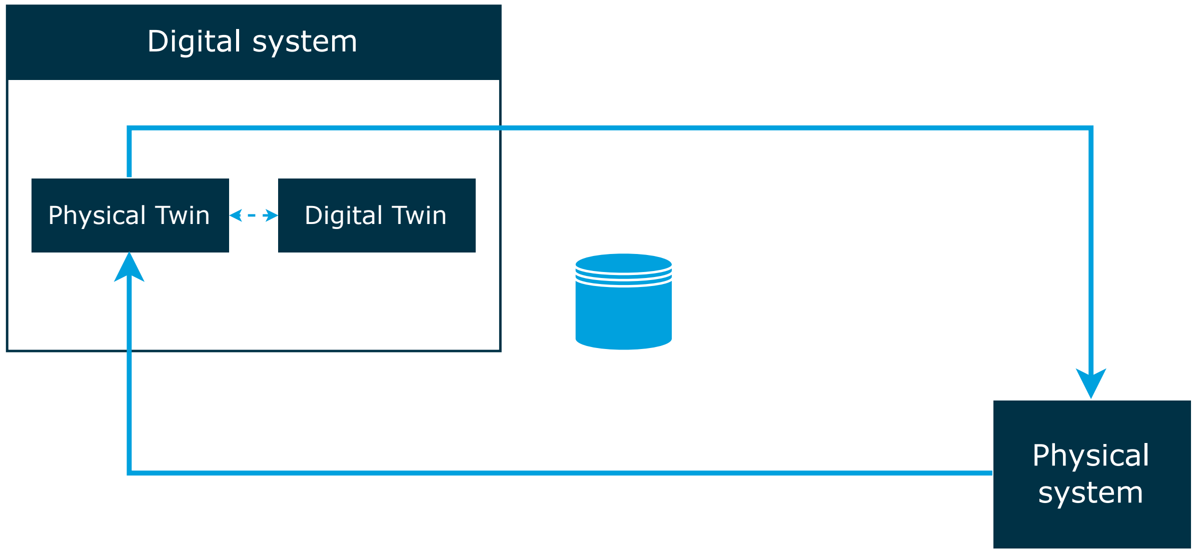


Figure 2.9: Digital Twin (DT) definition proposed in this study

In literature, the concept of a DT has received increasingly received attention. Tao et al. [110] and Ait-Alla et al. [23] have mentioned several benefits of DT application in supply chain management such as the possibility for remote and instant monitoring of operations and proactive risk and disruptions mitigation through timely decision making. However, the focus of applications of DTs has mainly been on manufacturing and according to Haße et al. [57], more research is needed with regards to logistics. Several studies have applied the DT concept to different fields. For example, Verdouw et al. [111] have applied a DT concept in order to advance smart farming through increased virtualisation. Hauge et al. [112] have developed a DT test bed in production logistics which utilises real-time location data. Stan, Borangiu, and Răileanu [113] have developed data-driven DTs for improved design and logistics control of product distribution by optimising palletising schedules, controlled with situational awareness and resource health monitoring. Furthermore, Hofmann and Branding [106] have developed a DT for truck dispatching operator assistance in port operations, which enables the determination of optimal dispatching policies using simulation-based performance forecasts. Based on the three stages of DT integration in Figure 2.7, a model may only be truly classified as a DT in the case of automatic data flow between the real system and the virtual model. However, as indicated by the work of Hofmann and Branding [106], the data flow from the DT into the real system is not always considered to be necessarily automatic. In this case, the DT may receive information from the real-world system, which is then utilised for additional functionalities such as improved decision-making support. Similarly, Coelho, Relvas, and Barbosa-Póvoa [108] have developed a DT module which receives information from the simulation and allows a decision-maker to interact with the system in real-time. Therefore, although a true DT may not be accomplished in this work with regards to the presented definition and dimensions, it does provide a first step towards the improvement of the pharmaceutical air freight cool chain by means of improved decision-making through a DT concept. The improvement has been quantified through the application in a case study, for which an introduction has been provided in the following section.

2.7. Case study company overview

One of the players in the air freight industry which might benefit from improving cool chain management through the application of the DT concept is KLM Cargo. Since a case study has been performed at this company, a brief introduction and overview have been provided as a background for the case study described in Chapter 3.

2.7.1. Facts and Figures

KLM has been founded in 1919 and is the oldest airline still operating under its original name today. KLM is headquartered in Amstelveen while its hub is situated at the nearby Amsterdam Schiphol Airport, typically referred to as Schiphol (SPL). KLM contains three core activities; passengers, cargo and engineering & maintenance. As of 2004, KLM has been merged with Air France to form the Air France-KLM Group. Together, the airlines fly to 318 destinations in 118 countries. A complete overview of the overall company structure has been provided in Figure 2.10. Due to the merger of Air France and KLM into the Air France-KLM Group, the respective cargo divisions have also been merged into Air France KLM Martinair Cargo (AFKLMP Cargo). KLM can be considered as a typical combination carrier which utilises remaining aircraft capacity with the transportation of freight. KLM Cargo is the GHA which arranges the air cargo transportation through the acceptance of bookings for flight, feeding of freight towards the hub and the storage, handling and preparation of freight. Therefore, the actual air transportation of cargo is performed by KLM. As of 2008, Martinair Holland N.V. has been incorporated into the Air France-KLM Group and from 2011 it has been transformed into an all-cargo carrier operating out of SPL. Although operations are still performed by Martinair, the airline can be considered as the all-cargo carrier of KLM. Furthermore, the handling of freight for this carrier has been contracted out to a separate GHA called Menzies. In 2017, the Air France-KLM Group was the ninth largest cargo carrier with a total cargo traffic of 8,583 Freight Tonne Kilometre (FTK), of which 4,843 FTK has been contributed by KLM.

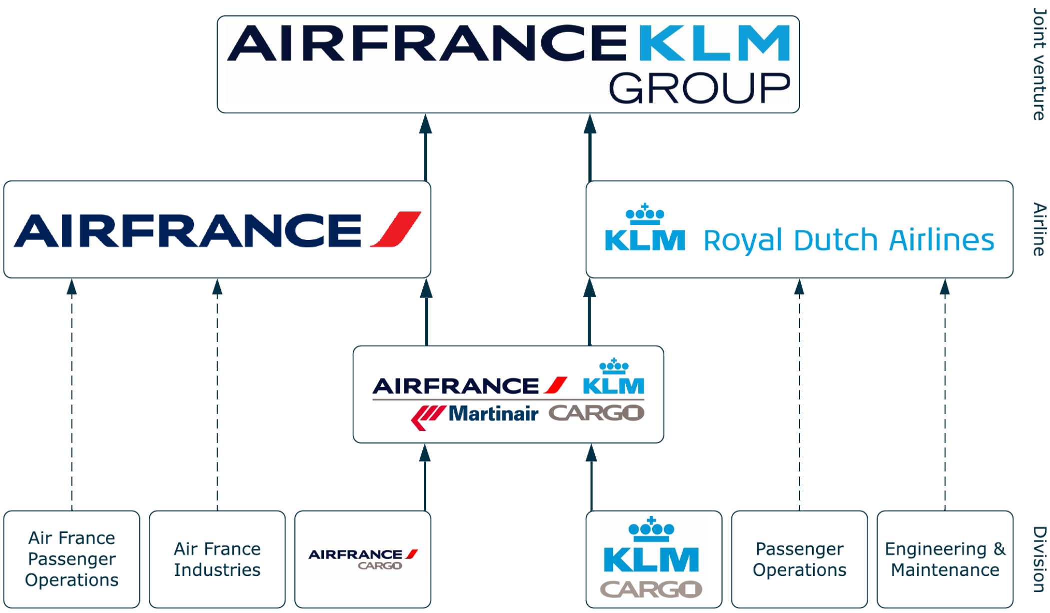


Figure 2.10: Company structure overview, adapted from Hensens [114]

From SPL, both KLM and Martinair serve an extensive worldwide network. Although subject to change, the combined network has been shown in Figure 2.11, highlighting the KLM and Martinair network in blue and red respectively. Since the operations of Martinair are not a part of the scope of this work, it has not been considered any further. In principle, the KLM network configuration is driven by the passenger side of the business where flight frequency might differ as well depending on the season. Besides the network shown in Figure 2.11, the network of Air France may be utilised as well which is operated from the Paris Charles-de-Gaulle (CDG) hub. It should be noted that the KLM network shown in Figure 2.11 only depicts the air freight destinations for Intercontinental (ICA) destinations. This is due to the fact that all inbound and outbound freight to and from European destinations is transported by trucks towards or from SPL.

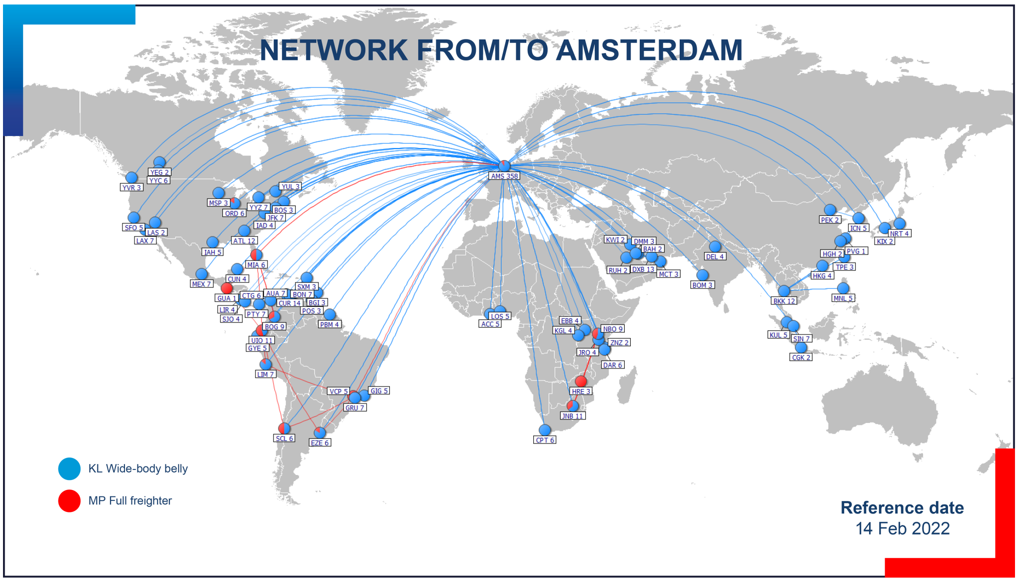


Figure 2.11: The operational network of KLM and Martinair out of SPL

2.7.2. The pharma variation

KLM Cargo arranges the air transportation of many varieties of cargo. However, only the pharmaceutical cargo variation has been considered as indicated in Chapter 1. In general, pharmaceutical shipments can be considered to be the most valuable from a business perspective as well as a shipment perspective. In fact, from the total AFKLMP Cargo revenue, roughly 25% can be accredited to pharmaceutical shipments. Furthermore, pharmaceutical shipments can be characterised by significant individual shipment values. On the one hand, the business value of pharmaceutical shipments advocates this segment as one of the most important areas of growth for KLM Cargo. On the other hand, regulations, high shipment values and the inherent sensitivity of such products necessitate the need for adequate care throughout the air freight logistics chain. Therefore, specialised products have been dedicated to pharmaceutical shipments, which can be seen in Table 2.2.

Solution Category	Vaccines	Closed Cool Chain Solutions			Controlled Cool Chain Solutions		
Product	Customised Covid Vaccines	Pharma Active Container	Pharma Passive Container	Pharma Hybrid Container	Pharma Control +2+8	Pharma Control +15+25	Pharma Control +2+25
SHC	SHL	ACT	ACE	PIP	COL	CRT	ERT
Product Code	c55-59	s/c52	s/c52	s/c54	s/c51	s/c53	s/c50
Temperature Range	Depending on requirements	-20 °C +20 °C	0 °C +30 °C	Depending on supplier	+2 °C +8 °C	+15 °C +25 °C	+2 °C +25 °C
Specifications	Depending on requirements	Dry ice operated	Electric	Hybrid			

Table 2.2: Pharma product group solutions

Referring to Table 2.2, three solution categories have been devised for the pharma variation: vaccines, closed cool chain solutions and controlled cool chain solutions. The latter two contain several different cool chain products for pharma offered by AFKLMP Cargo. Each pharma product has a dedicated Special Handling Code (SHC), which is used as an indicator for the specific requirements of a shipment. As an example, the COL indication on a shipment provides the specific temperature range required for safe handling: +2°C to +8°C. The product code a shipment has been booked with, determines the associated SHC.

Closed cool chain solutions

The closed cool chain solutions provide the possibility for customers to ship highly sensitive and valuable products without a break in the cool chain by use of an active, passive or hybrid container, also referred to as an active Unit Load Device (ULD). The active product is comprised of ULDs operating with dry-ice, the passive product is served by electrically powered ULDs while the hybrid product deploys

ULDs with a combined operating principle regarding the latter two. The benefit of the closed solution is the elimination of cool chain breaks since the shipment is in a controlled environment at all times throughout the logistics chain. Although such shipments are therefore under the strictest temperature control, special care is required in terms of temperature, battery and dry ice readings and or operational checks. Furthermore, the high cost of the closed cool chain solutions limits the application to each pharma shipment. The closed cool chain solutions have not been further considered since it does not fit the scope of the research project.

Controlled cool chain solutions

The controlled cool chain solutions include three temperature ranges in which shipments must be maintained at the warehouse, during air transportation and trucking. The first pharma control product is indicated by the SHC COL and specifies an allowed temperature range of +2°C to +8°C. The second control product is indicated by CRT with an allowable room temperature range of +15°C to +25°C. Finally, the third control product has been annotated with ERT, indicating an allowable extended room temperature range of +2°C to +25°C. Even though specific infrastructure such as cool storage has been arranged for COL and CRT shipments, tarmac and warehouse handling operations may temporarily expose shipments to ambient temperatures both at the hub as well as at the origin and destination. Therefore, shipments transported using any of these solutions are primarily exposed to ambient environmental conditions during cool chain breaks.

Vaccines

Recently, a new vaccine solution has been constructed in reaction to the SARS-CoV-2 outbreak in order to specifically serve the global Covid vaccination distribution campaigns. The temperature range and specifications of vaccine shipments highly depend on the type of packaging or ULDs used by the manufacturers, and are therefore case dependent.

2.8. Literature synthesis

In this chapter, a broad survey of cool chain management has been provided in which a general overview of air freight and its actors has first been provided. Then, imperative considerations for cool chain management have been discussed, from which it has become apparent that information extraction and improved decision-making may provide a suitable method for cool chain management improvement. Especially with regard to information extraction, it has been recognised that missing data is a frequently occurring issue in industry. Therefore, the different available methods for missing data imputation have been discussed. Thereafter, a novel performance indicator for the operational quality of a cool chain has been derived through an adaptation of the OEE metric: the OCCE. Subsequently, the different available modelling techniques for studying a system have been discussed, from which the DES technique has been chosen as most suitable for a logistics system such as a cool chain. Consequently, the DT concept which has been chosen to be utilised as the cool chain operational quality improvement method has been introduced. Furthermore, the DT development methodology has been introduced which consists out of the PT virtualisation model and the DT module. However, given the fact that this research has not been aimed at pharmaceutical cool chains in general, the chapter has been finalised through the introduction of KLM Cargo, at which the case study has been performed. Therefore, following the general description of cool chain management, in the following chapter the research has been narrowed down through a thorough description of the studied research object; the pharmaceutical air freight cool chain process at KLM Cargo.

3
KLM Cargo case description

In the following chapter, the KLM Cargo case description has been described in order to provide a sufficient process understanding while highlighting the current situation. Firstly, a brief overview of the Schiphol hub has been provided in Section 3.1, after which the studied process has been described in Section 3.2. Consequently, the encountered infrastructure and equipment in the studied cool chain process have been elaborated on in Section 3.3. Then, with a sufficient process description in mind, the system evaluation practices have been discussed in Section 3.4, by considering the data collection and used KPIs at KLM Cargo. The chapter has been finalised with a synthesis of the previously described findings in Section 3.5 in order to provide an answer to the following research question:

2 What is the current state of a pharmaceutical air freight cool chain process, based on an applied case study?

3.1. The Schiphol hub

AFKLMP Cargo operates from two distinct hubs: CDG and SPL. Only the latter hub is the base of operations for KLM Cargo and has therefore been further discussed. The hub at SPL provides several intermodal connections, which have been summarised in Figure 3.1.

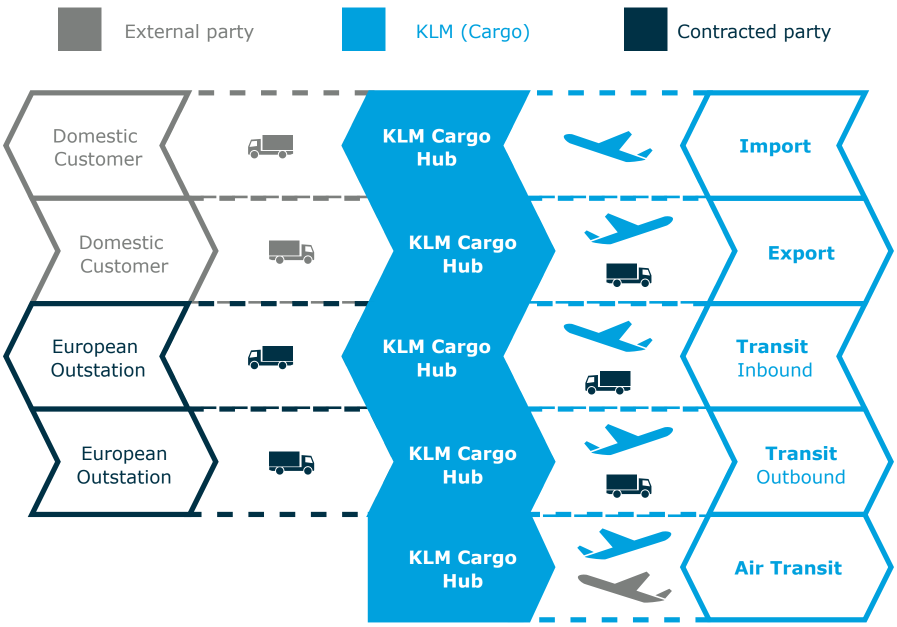


Figure 3.1: Intermodal connections and shipment flows at the KLM Cargo hub

In specific, there are four types of connections, of which two are intermodal; truck to truck, truck to aircraft, aircraft to truck and aircraft to aircraft. Combined, the four transportation connections serve three main shipment flows; import, export and transit. Referring to Figure 3.1, the transit flow can be divided into three flows; transit inbound, transit outbound and air transit. In principle, all European KLM Cargo destinations are serviced at outstations via trucks by contracted parties. Air transit shipments remain within the airport premises and are handled in order to continue the respective itinerary on a connecting flight. Furthermore, import shipments arrive at the Schiphol hub and are transported to a destination within the Netherlands. Finally, export shipments originate within the Netherlands and are accepted at the hub to be transported towards a European or ICA destination. Given the presented research scope in Section 1.4, only the transit outbound truck to aircraft flow has been considered.

Besides the different shipment flows, the handling at the hub is also influenced by the type of shipment configuration. The following freight configurations can be encountered:

- **Loose freight** includes shipments that are accepted at KLM Cargo which have not been prepared for flight by building it on a ULD. As an example, loose freight includes deliveries of loose boxes or wooden pallets, also referred to as skids.
- **Mixed-ULD (M-ULD)** is a freight configuration that contains multiple shipments built onto a ULD typically with differing destinations. A M-ULD has to be broken down after which the individual shipments are handled further at the hub and finally built up again on a ULD for flight.
- **Through-ULD (T-ULD)** is a shipment or combination of shipments which is already built on a ULD with one destination and is thus ready for flight. Therefore, a T-ULD does not require further handling apart from possible (cool) storage and transportation to the aircraft. Whenever a T-ULD has additional capacity left, shipments with the same destination might be added onto the ULD. However, this has not been considered in this research.
- **Active, Passive and Hybrid containers** are used for pharmaceutical shipments by certain customers choosing the closed cool chain solution. Such shipments are placed into the active, passive or hybrid ULDs by the shipper and thus only require limited handling by KLM Cargo. Nonetheless, such shipments should be monitored on for instance battery or dry-ice levels in order to ensure the correct transportation conditions.

With regards to the scope, only T-ULDs have been considered. Therefore, the remaining freight configurations have not been further discussed. A spatial overview of the three freight buildings at SPL for the specified flow and freight configuration has been provided in Figure 3.2. Cargo enters through freight building three and proceeds through the Pallet Container Handling System (PCHS), which is situated throughout freight buildings two and three. A more detailed overview of the PCHS system has been provided in Section 3.2. The cool cell which has been incorporated into the scope is situated in freight building one, which is accessed from the air side. Furthermore, in front of freight building two and three are multiple air side lanes, at which cargo is placed awaiting transportation towards the aircraft.

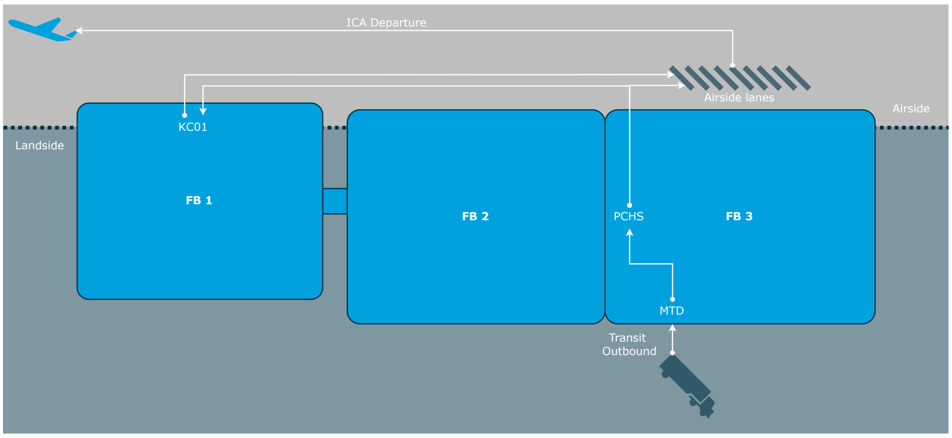


Figure 3.2: Spatial overview of the three KLM Cargo freight buildings at SPL

3.2. Process description

For the transit outbound T-ULD flow, the process of transporting freight can be divided into three parts; feed, handle and distribute. The feeding constitutes the road transportation of T-ULDs from the European outstations towards the hub at SPL. The handling is performed at the hub and includes the unloading, storage and movement of the freight. Finally, cargo is distributed throughout the KLM network by aircraft. Although some issues may arise during trucking or flight, in principle pharmaceutical freight is transported according to the correct conditions. Therefore, from a pharma cool chain perspective, this work has been focused on the handling at the hub, since this part contains the most significant cool chain breaks and thus constitutes to any environmental exposure most significantly. The corresponding process map has been shown in Figure 3.3.

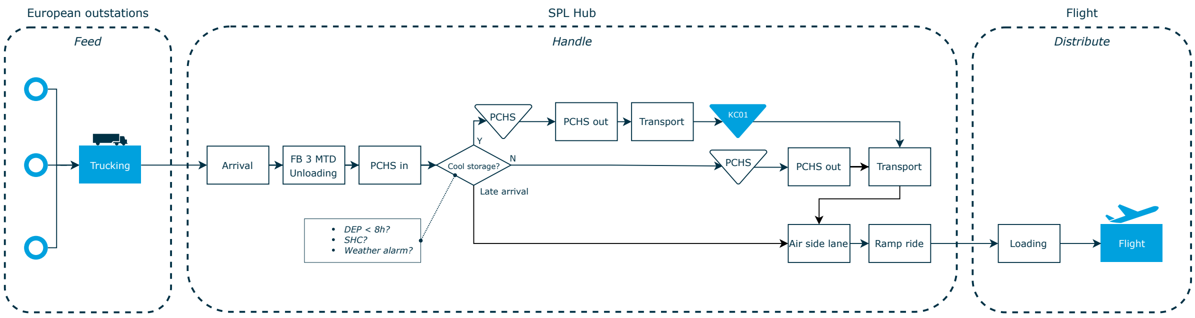


Figure 3.3: Process map for transit outbound T-ULD handling at the SPL hub

Considering the flow chart shown in Figure 3.3, it has been clarified what actually flows through this process. In the airfreight industry, a shipment which has been booked at an airline such as KLM in this case, is represented by the Air Waybill (AWB). The AWB is a contract of carriage between the shipper and the carrier which includes detailed information regarding the colli. As an example, an AWB might indicate that the shipment is comprised of ten boxes, the number of colli, which are to be transported under a specific product code, which provides necessary information on the required temperature range. Each booked shipment consisting possibly out of multiple colli, receives a unique AWB with the corresponding product code and thus SHC. However, since one pallet ULD may be able to hold more than one AWB, it is possible that one ULD may be holding multiple AWBs. Vice versa, the colli of one AWB may be distributed over several ULDs. The decomposition of shipments has been visualised in the schematic shown in Figure 3.4. For clarity, the terms shipment and AWB and freight unit and ULD may have respectively been used interchangeably throughout this work.

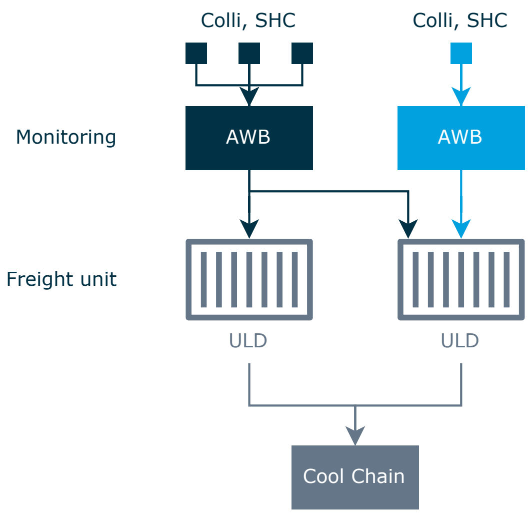


Figure 3.4: Decomposition of shipments flowing through the system

It has been noted that in the case of multiple AWBs sharing one ULD or freight unit, the SHC should be equal for both shipments. Furthermore, the individual colli which make up a shipment are not traceable throughout the KLM Cargo cool chain. Any monitoring which is performed is only performed on AWB level, to which unique ULDs may be coupled. Finally, the freight units are the actual physical units flowing through the system as shown by Figure 3.3.

The analysed process is initiated upon arrival of a truck at the hub and completed once a flight departs, which is also referred to as the off-blocks time or the time an aircraft is pushed back from the gate for departure. In between, the process has been roughly divided into four parts: arrival, PCHS, cool storage and ramp ride and loading. Throughout these processes, there are several hand-overs from department to department responsible for the freight. Upon arrival of the freight, the Import & Export department arranges the unloading of the trucks and the entrance into the PCHS. Then, once the freight has entered the PCHS, the ULDs are handed over to the Transport department, which controls the movements of the freight until loading into the aircraft. The Freight Control Center is part of the Transport department and plans and arranges all movements of ULDs within the PCHS. Finally, after delivery to the aircraft on the ramp, the freight is handed over from Transport to Ground Services, which is not part of KLM Cargo.

3.2.1. Arrival

As determined in the research scope, the studied flow contains only freight that is accepted at European outstations and fed towards the hub for further processing and distribution. Besides, only the T-ULD configuration has been considered. Therefore, each freight unit arriving into the considered system is comprised of a ULD which is in principle checked and configured for flight. The road feeding trucks from European origins are operated as flights and thus have a Scheduled Time of Arrival (STA). Since the arrival time of trucks might differ due to traffic jams or technical breakdowns, the Actual Time of Arrival (ATA) at the hub has been taken as the moment when freight enters the system. After the arrival of a truck, it typically needs to wait before unloading can commence. When required, a truck trailer is in principle fitted with a cooling system which is able to provide the necessary temperature setting for the T-ULDs which are on board. Each truck can contain four to six ULDs, which may or may not be bound for the same connecting flight. Once possible, the truck docks at the Moving Truck Door (MTD), which contains a transport vehicle which is able to unload one full ULD at a time. In the case of a late arrival, which is at the latest six hours before the Scheduled Time of Departure (STD), the ULD is not entered into the PCHS system. Instead, it is placed onto a dolly and driven by a tractor towards the air side lanes to await further transportation towards the aircraft.

3.2.2. PCHS handling

After the arrival process, each T-ULD enters the PCHS; the automatic storage and retrieval system for pallets and or containers. Upon entry of a ULD after the MTD, it is handled almost fully automatically through a series of elevators, Automated Guided Vehicle (AGV), turning tables and conveyor belts. The system is situated above the warehouse floors of freight buildings two and three and it is, therefore, possible for ULDs to traverse the buildings from landside to airside without significant interruptions. Upon entry of the PCHS, a decision is made for each pharma T-ULD whether to place it in cool storage or not, depending on three business rules:

- **DEP < 8 hours**: the commercial promise to shippers is a maximum exposure time to environmental conditions of eight hours while a shipment is handled at the hub. Therefore, any shipment with a transit time of less than eight hours is not placed in cool storage.
- **SHC**: In the case that a shipment has a transit time exceeding eight hours, it is only placed in cool storage in the case of a SHC COL or CRT, indicating the respective required temperature range and thus cool storage.
- **Weather alarm**: in the case that a weather alarm is issued when the ambient temperature exceeds 18 °C or is below 5 °C, shipments with SHC ERT are stored in the CRT cool storage when the transit time exceeds eight hours.

Any shipment which is not placed into cool storage according to the previously described business rules is temporarily stored in the PCHS system, awaiting the departure of its respective connecting flight. In principle, the described business ruling for cool storage has thus been considered of static nature, since the respective decision does not depend on the current system state in terms of for example encountered temperatures throughout the process.

3.2.3. Cool storage

In the case that a shipment requires cool storage, it is removed from the PCHS at the air side and transported towards freight building one, at which the cool storage for T-ULDs is situated, also referred to as KC01. The cool storage exists out of two environmental zones; one for COL shipments and the other for CRT and ERT shipments in the case of a weather alarm. In principle, storage and warehousing is not the core business of KLM Cargo. However, in order to reduce cool chain breaks, especially in the case of freight with long transit times at the hub, cool storage is used in an attempt to reduce freight exposure to ambient environmental conditions.

3.2.4. Ramp ride and loading

Before the transportation towards the aircraft, or ramp ride can take place, a shipment is removed from storage in either KC01 or the PCHS. Since each ULD is taken from storage individually, it is firstly placed at the so-called airside lanes, shown in Figure 3.2. These lanes act as a buffer to collect and place ULDs booked on the same flight on so-called dollies, before being transported by a pulling tractor towards the aircraft. The air side lanes are situated in front of freight building two and three and are fully exposed to ambient environmental conditions. At maximum, a tractor can pull five dollies during a ramp ride. Once the tractor arrives at the departure gate, the ULDs are handed over to Ground Services, which arranges the loading of the aircraft. During loading, ULDs may experience prolonged exposure to ambient temperature and the aircraft cargo hold is not conditioned until the engines are running after off-blocks time.

3.3. Infrastructure and equipment

In this section, an overview has been provided of the respective cool chain infrastructure and equipment for the given case study, along with relevant technical details.

3.3.1. ULDs

Although a significant number of ULDs are used in the air freight industry, for the studied T-ULD flow primarily two types are used; the P6P-PMC and P1P-PAG pallets. The ULDs have been shown in Figure 3.5.

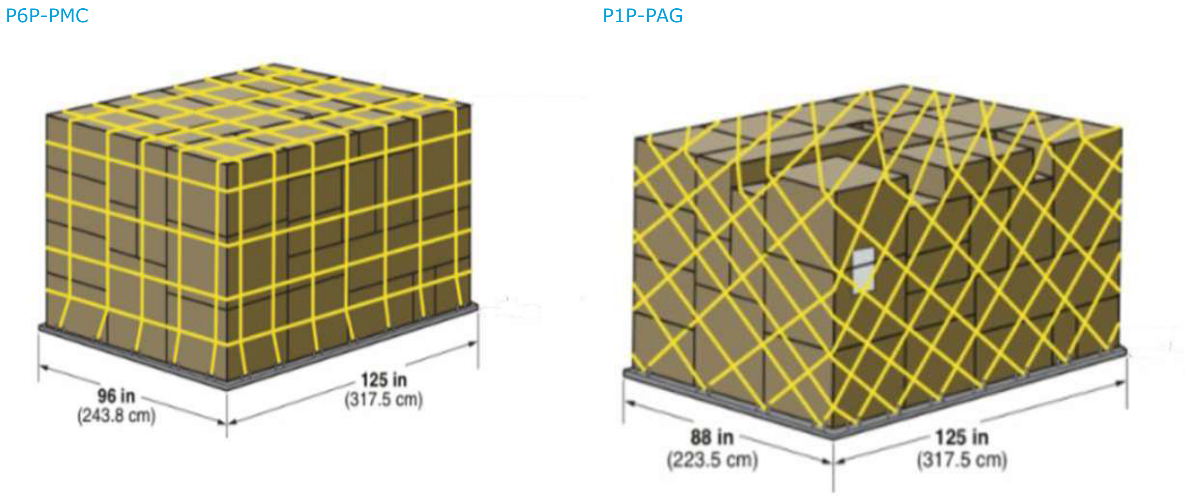


Figure 3.5: ULD types

The ULDs are comprised of reinforced metal sheets on which cargo is built and stacked, at the outstations in the case of T-ULDs. The freight is secured onto the pallets by means of netting, which is attached to the edges of the ULDs. Given the significant size and weight of these freight units, specialised equipment is utilised in order to handle them.

3.3.2. MTD

The MTD is an Electric Transport Vehicle (ETV) operated by two persons and is capable of unloading one ULD at a time. On the land side of the freight building, an open door is situated at which an incoming truck is able to dock. After docking, ULDs are rolled out of the trailer onto the MTD one by one for further processing. As mentioned before, depending on the arrival time of the freight, ULDs are either entered into the PCHS system through an automatic lift or are placed on dollies for transportation to the air side lanes in the case of a late arrival. In principle, the time it takes to unload and process one ULD is roughly 7.5 minutes.

3.3.3. PCHS

The PCHS is the automatic storage and retrieval system which spans both freight building two and three. The system provides 1,749 temporary storage places for ULDs awaiting transportation towards the aircraft, as well as the transportation throughout the freight buildings. In principle, movements are automatically coordinated by the system, although personnel is able to intervene manually. On average, it takes up to 15 minutes for a ULD to enter the PCHS system. Upon entrance into the system, the decision-making for cool storage is performed by the WMS Chain, according to the indicated business rules. Furthermore, the system is able to retrieve a ULD from a storage position and place it outside in an average of 5 to 10 minutes. Since ULDs which do not go into cool storage are stored in the PCHS, removal should be performed in time in order to transport it on time towards the aircraft. The time of removal is determined by the system according to the master flight table, which contains all scheduled flights, the days on which they depart as well as the STD. An example overview of a flight shown in the master flight table has been provided in Figure 3.6.

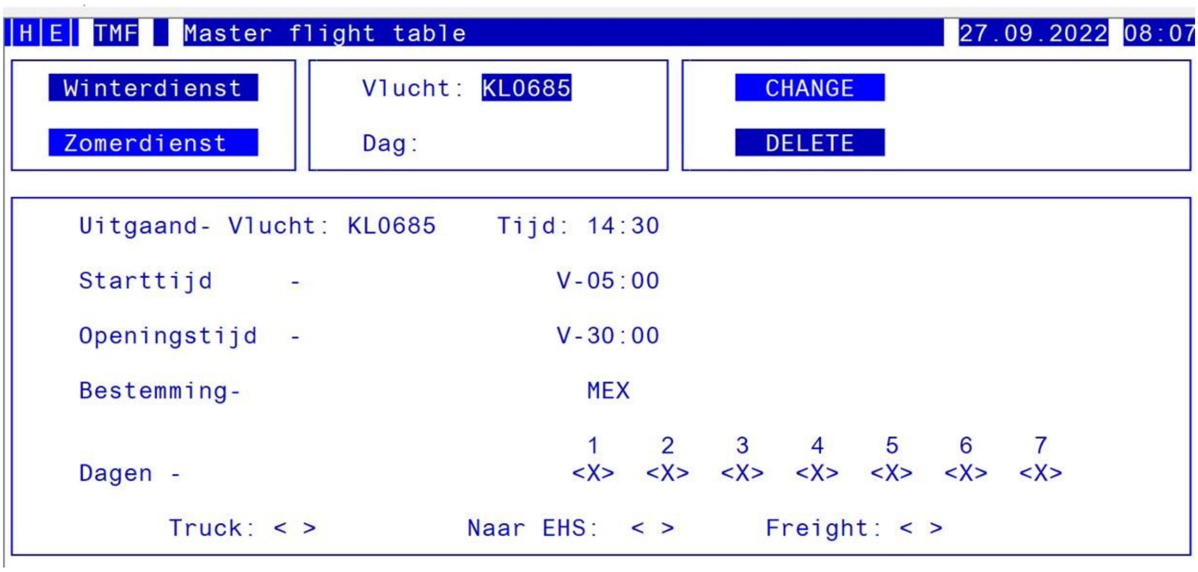


Figure 3.6: Example screen of the master flight table

An operational rule depicts that a ULD is removed from the system at 5 hours before STD, or V-5 hours. The times indicated in the master flight table are fixed. However, if necessary, personnel is able to alter the removal time in the current flight table.

3.3.4. KC01

The cool storage for T-ULDs, or KC01, has a cool room for CRT and ERT ULDs and a cold room for COL ULDs. The set point for the cool room is set at 20 °C while the set point for the cold room is set at 5 °C. Although small fluctuations in temperature may occur, in principle the control system of the storage rooms ensures the correct storage temperature at all times. The CRT cool room provides a total maximum capacity of 33 ULDs, whereas the COL cold room provides a maximum storage capacity of 42 ULDs. However, it should be pointed out that this is a maximum capacity, which is also utilised for fresh ULDs, besides the pharmaceutical freight. Furthermore, different ULD flows such as aircraft-aircraft transfers may require cool storage at KC01. Therefore, data analysis has been performed and discussed in Section 4.1 in order to adjust the storage capacities to the research scope. As mentioned in Section 1.2, KC01 is characterised by capacity constraints. Moreover, the cool storage contains only one entry and exit point for ULDs which is serviced by a single ETV, shown in Figure 3.7.

Figure 3.7: The ETV storing or retrieving a ULD in KC01

The limited speed of the ETV together with the single point of entry and exit provide an additional challenge in providing the required care for pharma ULDs. The ETV is able to process around 6 ULDs per hour, which results in an average processing time of 10 minutes per ULD. Operationally, the rule has been set that ULDs exiting KC01 have priority over ULDs entering the cool storage. Ideally, ULDs are removed at the earliest three hours before the STD in order to limit any exposure to ambient conditions while allowing for timely handling. However, given the significant processing time of the ETV, the removal time may be more than three hours before STD in order to ensure all outbound ULDs are removed on time.

3.3.5. Transporter

For any freight movements around the three freight buildings at the hub, specialised transporters are used at KLM Cargo, shown in Figure 3.8. A transporter can handle a single ULD at a time by loading to or unloading it from roller beds situated at the entrance and exit positions at both the PCHS as well as KC01. Furthermore, transporters are used to place ULDs on the dollies which are used for the ramp ride towards the aircraft. For transportation to and from KC01, there is always one dedicated transporter at the hub. For other movements, such as from the PCHS exit to a dolly on an airside lane, usually in total five transporters are scheduled. Similarly to the KC01 storage capacity, however, these transporters are also used for any other ULD and flow.

Figure 3.8: Transporter vehicle at KLM Cargo

3.3.6. Tractor

Tractors are used at KLM Cargo for the ramp rides to and from aircraft, as well as the movement from the MTD to an airside lane in the case of a late arrival. In total, there are on average 30 tractors available at the hub. Each tractor has a maximum towing capacity of five dollies, which are the carts on which ULDs are transported during ramp rides.

3.4. Performance management

Throughout the studied system, data is generated, collected and used to determine the system performance by means of KPIs. In the following subsections, firstly the data collection has been discussed after which the relevant KPIs in use at KLM Cargo has been considered.

3.4.1. Data capture

One of the main sources of operational data at KLM Cargo is from the WMS Chain. In principle, Chain is the system which provides the main coordination of the hub operations. As an example, the system provides personnel with transportation tasks of freight. Moreover, the system automatically collects vast amounts of data for each AWB on a ULD level for the given scope. An example of collected data includes for instance time stamps at certain locations of a specific ULD in the system. Moreover, crucial information such as the corresponding AWB number and SHC is also recorded. Despite the automatic data collection, manual adjustments can be made by personnel in the case that any data is not correctly registered. Furthermore, some data is manually entered into the system upon completion of a process step. Throughout the studied system, there are multiple points at which timestamps have been recorded for each individual ULD. An overview of the timestamps has been shown in Figure 3.9.

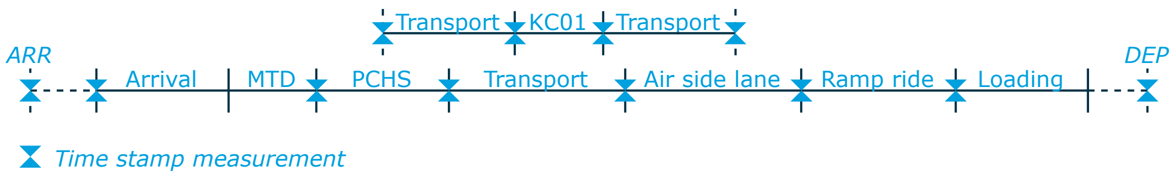


Figure 3.9: Timestamp measurement points

Timestamps are not recorded at the beginning and end of each processing step. As an example, although the ATA is recorded for each truck and thus ULD, the moment at which the truck docks at the MTD is not recorded. Similarly, the time at which unloading by the MTD commences is not recorded. Therefore, the total time of the arrival process until a ULD enters the PCHS, includes the processing time of the MTD. Moreover, regarding the loading process, only the time at which a ramp ride has ended is known. However, since it can be assumed that a ULD is situated in ambient environmental conditions until departure, it is not necessary to utilise the loading time with regard to temperature exposure.

Besides the process timestamps, temperature sensors have been installed throughout freight buildings two and three. The sensors and the corresponding temperature readings are connected to and shown on the online ATAL web platform. Currently, the ULDs at KLM Cargo have not been fitted with smart tags or Internet of Things devices in order to extract environmental data from the immediate surroundings of the cargo. Therefore, the exact conditions in which freight is handled are only known to a certain degree. However, the temperature data available from this system is not utilised in terms of decision-making with regard to cool storage.

3.4.2. Key Performance Indicators

At KLM Cargo, a range of KPIs are applied in order to measure the operational quality of the freight processes. Similarly, for the presented case study, multiple KPIs have been specified. Generally speaking, the operational quality of the KLM Cargo cool chain is twofold. On the one hand, pharmaceutical freight must be handled in time in order to be flown according to the schedule or booking. On the other hand, as readily discussed, the freight should be handled under the indicated temperature conditions as much as possible. The KPIs which are used at KLM Cargo have been further discussed separately.

Handling deadlines

In general, the timeliness of cargo handling has been represented through several handling deadlines, which are specific times at which a ULD is supposed to be at a given place within the process. For the given scope, there are three handling deadlines:

- **Handling deadline 1**: ULD received on time into the MTD process. On time has been further defined as the minimum connection time plus 60 minutes landside time, where the minimum connection time is 300 minutes before STD. Therefore, handling deadline 1 is met when a ULD is received into the MTD at the latest six hours before STD.
- **Handling deadline 2**: ULD handed over on time from MTD to Transport. On time has been specified as the minimum connection time plus 60 minutes landside time plus 30 minutes processing time for the MTD, where the minimum connection time is 270 minutes. Consequently, a ULD is handed over on time to Transport when it is done at the latest six hours before STD.
- **Handling deadline 3**: ULD handed over on time from Transport to Ground Services. A ULD is handed over on time to Groud Services at the latest 80 minutes before STD. The deadline thus ensures sufficient time for loading the freight onto the aircraft before the respective scheduled departure time.

Each deadline is boolean in nature since a deadline is either met, indicated by *OK*, or not which is indicated by *NOT OK*. Although a missed deadline does not necessarily mean that a ULD will miss its connecting flight at the hub, it does indicate a deviation from the required handling as set by management.

Cool chain deadlines

Similarly to the handling deadlines, three deadlines have been formulated specifically for the cool chain. Similar to the handling deadlines, the cool chain deadline KPIs are boolean in nature. The deadlines have been elaborated on in further detail:

- **Cool chain deadline 1**: a ULD enters the PCHS within 120 minutes after ATA.
- **Cool chain deadline 2**: ULDs with a transit time of more than or equal to eight hours enter KC01 within 180 minutes after ATA.
- **Cool chain deadline 3**: a ULD which has been stored in KC01 is removed no longer than 180 minutes before STD.

Flown as Planned

Although the deadlines provide an insight into the handling process, a ULD with its corresponding AWB(s) may still be flown on the booked flight. Therefore, especially with the shipper in mind, the KPI Flown As Planned (FAP) has been used in order to indicate the degree to which the freight is flown on time. In specific, FAP has been determined as the ratio of AWBs which have been flown on the booked flight to the total number of AWBs. Since there is a significant range of reasons for a ULD to miss the scheduled flight, an appropriate scope should be set with regard to the performed research. As an example, a technical fault in an aircraft may negatively influence the FAP KPI, yet has not been taken into account. The assumptions which have been made have been further discussed in Section 4.2.4.

Time Out of Refrigeration

The Time out of Refrigeration (TOR) is used as a metric in order to quantify the operational quality with regards to the temperature aspect. The TOR specifies the time a ULD was not in a dedicated cool storage facility, in this case, KC01. In other words, the TOR can be determined by subtracting the time in refrigeration from the total processing time at the hub. KLM Cargo strives for a maximum TOR of eight hours, which is also the commercial promise to customers. Contrary to the deadlines, the TOR is measured on AWB or shipment level since one AWB may be distributed over multiple ULDs. Therefore, if one of the corresponding ULDs has a TOR of more than eight hours, the corresponding AWB is considered NOT OK in terms of TOR. It should be noted that currently, KLM Cargo is not able to steer the operational processes through the TOR metric. Therefore, it is solely recorded for performance measurement. Moreover, a more general notion of TOR can be defined as the amount of time that a product is outside of the specified temperature range according to the product code and SHC. Nonetheless, such information is currently not captured at KLM Cargo through the direct monitoring of ULDs or the digitisation of such information from other data sources. Therefore, there is no insight into the actual exposure of pharma T-ULDs, besides the time spent outside of KC01.

Cool storage availability

Given the KC01 capacity constraints, it has been deemed imperative to maintain a proper overview of the availability of the cool storage facility. At KLM Cargo, the number of ULDs stored in KC01 is recorded and displayed in a dashboard shown in Figure 3.10.

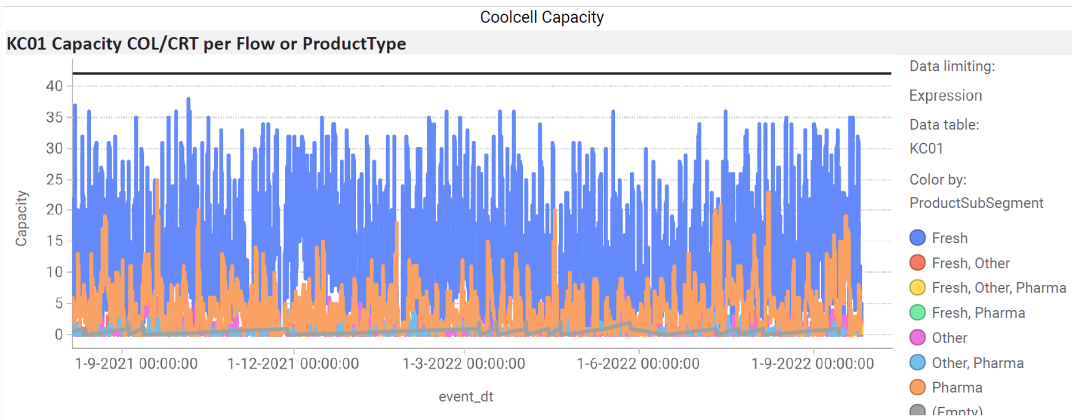


Figure 3.10: KC01 capacity dashboard

Besides the dashboard, which primarily provides insight into the historical storage capacity, personnel has access to information on the amount of ULDs stored in KC01 at a given time. However, there is no metric used in order to quantify the availability of the facility for further decision-making, which is primarily done ad hoc through operator experience.

3.5. Case study synthesis

The research object has been described in this chapter by considering a general overview of the airport hub and the studied cool chain process. Then, the infrastructure and equipment have been separately discussed, where most importantly significant capacity constraints for the KC01 cool storage facility have been noted. However, the business ruling in place for the determination of cool storage can be considered static, not taking into account for instance the temperature in another suitable storage location such as the PCHS. Besides, the current performance management has been discussed in terms of data capture and KPIs. With regard to data capture, it has been noted that although significant amounts of data are theoretically captured in the system, this data is not used directly with respect to real-time decision-making. Additionally, a significant number of KPIs are used in order to assess the performance of the system. However, the described KPIs do not provide a coherent overview of the operational quality of the system as a whole. Besides, there is no KPI in place which captures the temperature exposure on a ULD level. Consequently, it has thus been concluded that currently, the system is under strain from cool storage capacity restrictions, while data and information extraction is not utilised in improved decision-making for increased operational quality. Therefore, the studied and presented cool chain system may benefit from the application of the techniques and methods introduced in Chapter 2. However, in order to apply the DT improvement concept to alleviate the encountered complications, the required modelling has first been developed in the following chapter.

4 Model development

In the following chapter, the model development has been discussed by firstly considering the collection, handling and analysis of relevant data in Section 4.1. Thereafter, an outline of the model structure has been provided in Section 4.2. The chapter has been built up to provide an answer to the following research question:

3 How can a pharmaceutical air freight cool chain be modelled?

4.1. Data analysis

The first step of developing a simulation model in order to carry out a research project is the collection, handling and analysis of data. Firstly, any data which is relevant to the research subject should be collected. Consequently, the collected data should be handled in order to deal with missing or incorrect entries. Furthermore, the collected data should be prepared for usage in the model development. Finally, the collected and handled data can be analysed and utilised in the model development. Each step has been further clarified in the following subsections.

4.1.1. Data collection

For the analysis of the presented case study and thus the analysis of the research object, relevant data has been collected. In specific, historical data has been gathered from the WMS Chain. The department of Performance Management has provided a cool chain database with historical entries spanning from January 1st 2021 to January 1st 2022. In order to consider only entries relevant to the case study, the data set has been filtered in order to contain only the pharma T-ULDs during the indicated time frame. Each row in the data set contains information about the shipment, route information and process timestamps, as shown in Figure 4.1. From the data set, two respective data sets have been created from the parameters of the cool chain data set; an input data set and an analysis data set. As the name suggests, the input data set has been used as model input, whereas the analysis data set has been used in order to derive the modelling parameters.

Cool chain data set
Shipment information
Route information
Process timestamps

Input data set	
Label	**Example**
Product_code	S51
AWB_Origin	BCN
AWB_Destination	GRU
SHC	COL
ULD_In	PMC23246
Flight_In	KL8680
Flight_Out	KL0791
Actual_In	2021-01-01 04:08:00
Sched_Out	2021-01-03 09:55:00
Number of AWBs	2
AWBs	[7441760762 7441760773]

Analysis data set	
Label	**Example**
Freight_Unit	07441760762PMC23246
SHC	COL
Sched_In	2021-01-01 03:30:00
Actual_In	2021-01-01 04:08:00
Sched_Out	2021-01-03 09:55:00
Actual_Out	2021-01-03 09:54:00
PCHS_In	2021-01-01 04:28:00
PCHS_Out	2021-01-01 06:20:00
RR_Start_In	2021-01-01 06:25:00
RR_End_In	2021-01-01 06:28:00
Cool_In	2021-01-01 06:40:00
Cool_Out	2021-01-03 03:00:00
pu_lane_dt	2021-01-03 03:28:00
RR_Start_Out	2021-01-03 07:47:00
RR_End_Out	2021-01-03 07:59:00

Figure 4.1: Process related data set description

With regards to temperature, data has been collected from within KLM Cargo as well as from a publicly accessible source. For temperature readings within the warehouse, historical data from the ATAL temperature sensors system has been collected. Readings from a weather station at Schiphol of the Koninklijk Nederlands Meteorologisch Instituut have been collected for ambient or outside temperatures. The latter data set provided hourly temperature readings, which was the reason for also extracting warehouse temperature measurements on an hourly basis. Both data sources have been combined into a single temperature database that spans the same time period as the operational data. The combined temperature data set can be seen in Figure 4.2. For illustration purposes, only readings of one sensor have been shown. Furthermore, the label T represents the ambient temperature and all readings are shown in °C.

Temperature data set		
datetime	*Temperature reading sensor #*	*T*
2021-05-20 15:00:00	18.78	14.2
2021-05-20 16:00:00	18.893	14.2
2021-05-20 17:00:00	18.749	13.1

Figure 4.2: Temperature data set description

4.1.2. Data handling and preparation

As mentioned in Section 2.3, data sets in industry are typically subject to missing data. The issue of missing data has also been encountered in the cool chain data set and the temperature data set. The handling and preparation for each data set have been further elaborated on separately.

Cool chain data set

Given the occasional manual input and adjustments in the data coming from the Chain system, the reliability of the data is not optimal. Therefore, erroneous entries were expected in the data set. In fact, it has been observed that the quality of the cool chain data set is poor, which has been deemed as an area of improvement for KLM Cargo. Indeed, several issues have been found, mainly related to the process timestamps. Besides missing entries, negative process duration times have been encountered. It should be noted that the cool chain data set should contain missing values in the case that a ULD has not gone into cool storage, since in this case *Cool_In* and *Cool_out* have not been recorded. In order to prepare the data set for utilisation, several steps have been taken:

1. **Duplicate entries**: as a first step, the overall data set has been checked for duplicate entries. Since each AWB shipment may be comprised of multiple ULDs, duplicate entries with regards to the AWB number were expected. However, duplicate rows with equal AWB and ULD codes have been found. Such duplicate rows have been removed in order to ensure that the data set only contains unique entries.
2. **Processing times**: corresponding to the chronological order of processing steps and timestamps, the processing times for each part in the system have been assessed. Consequently, several processing times with negative or excessive duration have been found. Furthermore, this step also revealed missing time steps. Moreover, during the processing times verification, it has been found that the data set contained multiple DateTime data formats, leading to incorrect conclusions. After this issue has been addressed, the processing times have been assessed again.
3. **Missing and incorrect entries**: the obtained missing and incorrect time stamps have been further investigated. In the case that a missing or incorrect entry could not be resolved manually by considering similar shipments, the respective data entry has been omitted. Likewise, incorrect time stamps have been removed before the analysis of the data.

As a result of the data handling and preparation steps, a complete cool chain data set for the specific scope has been obtained. The overall data set has been structured on AWB level with the corresponding ULDs. However, for the input data set, it has been deemed more suitable to structure it based on ULD level, since the ULDs are the actual physical objects flowing through the cool chain. Therefore, a transformation has been performed while removing data entries which are not relevant to the model input data set. A schematic overview of the transformation has been shown in Figure 4.3. It should be noted that not all relevant parameters in the data sets have been shown in this visualisation.

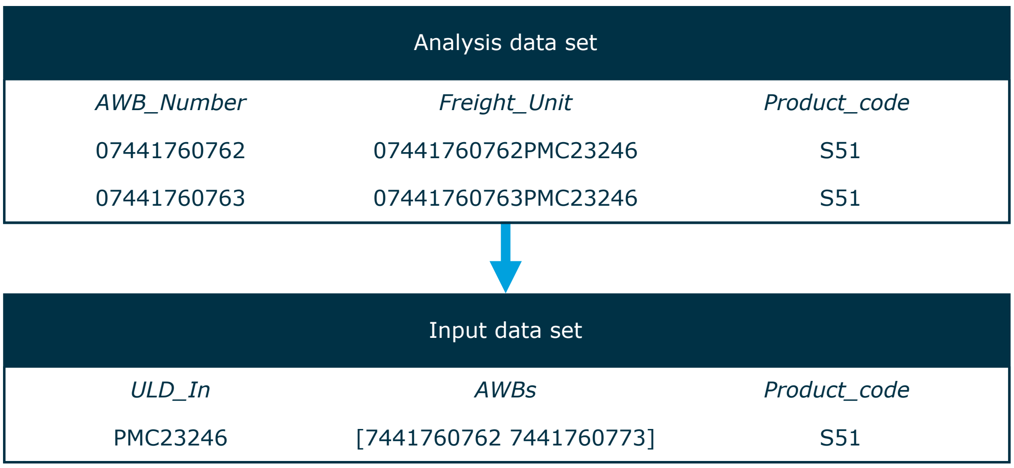


Figure 4.3: Transformation applied for the creation of a model input data set

After the data handling and preparation, the analysis data set contained a total of 4,937 unique rows, whereas the input data set contains 3,912 unique ULDs. The latter data set is smaller in size since a significant amount of AWBs share a ULD with a different shipment, or more correctly AWB number.

Temperature data set

In contrary to the missing values in the cool chain data set, the temperature data set contains measurement samples of a continuous variable, namely temperature. Therefore, in principle any missing value is erroneous. In order to provide insight into the missingness of the temperature data set, the missing data pattern has been visualised in Figure 4.4. The visualisation provides an overview of the temperature data set with as columns the different sensors and the total number of rows up to 8833, the total data set size. Therefore, each blank spot in the visualisation represents a missing data entry in the respective column in the respective row. It has been noted that for the case study, sensors 22, 24, and 71 have been used for the PCHS exit area, storage area and entrance area respectively.

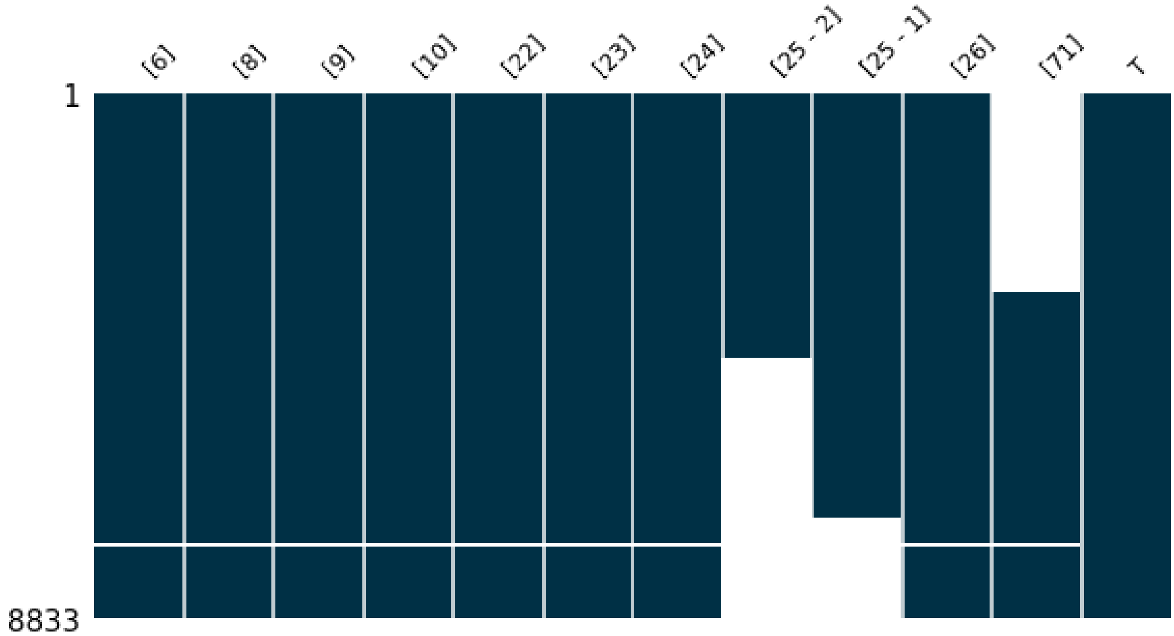


Figure 4.4: Missing data pattern of the temperature data set

Referring to the missing data patterns and mechanisms discussed in Section 2.3, from Figure 4.4 it can be concluded that the data set includes both a general missing data pattern as well as a line pattern where all sensors are missing values. Furthermore, in accordance with the responsible party at KLM Cargo, it has been concluded that the missing values have occurred due to sensor breakdowns or lost data transmissions. Therefore, since the missingness is only related to external causes and the measurements of other sensors have no influence on the blank values, the missing data can be considered MCAR.

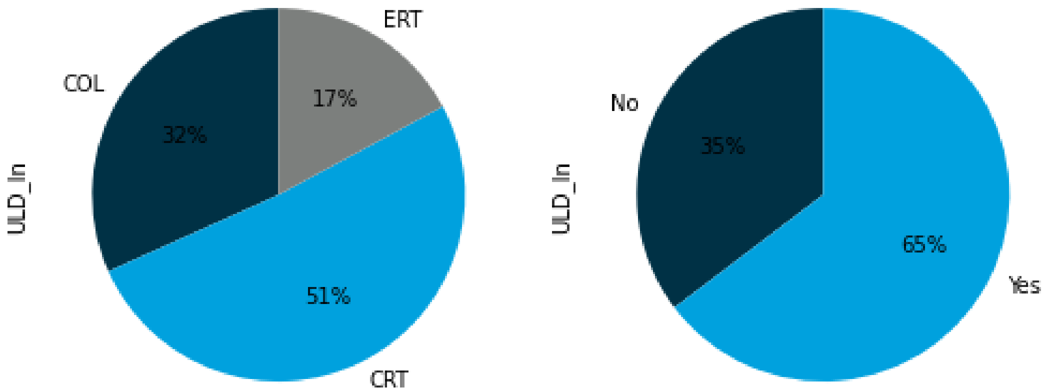
Since the temperature data concerns a time series, simply removing entries is not possible. Therefore, data imputation has been performed in order to handle the missing temperature measurements. In principle, manual data imputation using statistical means such as mean imputation is unsuitable due to the relatively long periods of missing values in the time series data set, as seen in Figure 4.4. Besides, neural network-based methods have been deemed too complex for the given research scope due to the required hyperparameter tuning. Therefore, model-based data imputation has been applied. Since EM imputation requires a significant data set size [115], i.e. typically around 500,000 data samples [116], the kNN data imputation method has been chosen. Furthermore, kNN missing data imputation is more suitable for medium-size data sets and the data requirements are more flexible [117]. For the kNN imputation method, a choice has to be made for the number of neighbours k, which practically entails the number of other data values which are used in order to estimate a missing value. From the literature, it has been found that $k = 3$ is a reasonable choice for proper missing value estimation while reducing the risk of impairing the variability of the data set [118]. The kNN imputation algorithm is readily available from the sklearn python package and has been used in order to impute the missing values shown in Figure 4.4. After imputation, the temperature data set did not contain any missing values.

4.1.3. Analysis

The cool chain analysis data set has been used in order to perform several analyses with the aim to derive system parameters for the development of the model. The analyses have been further discussed separately.

Freight characteristics

In order to provide a basic overview of the truck-aircraft flow, several characteristics have been analysed from the cool chain data set. In Figure 4.5, an overview has been provided on the general characteristics of the freight. For more than half of the recorded ULDs, the required temperature SHC is CRT, as seen in Figure 4.5a. Furthermore, only 17 % of the ULDs has the SHC ERT. Therefore, the majority of ULDs, or 83 %, is subject to strict temperature conditions. Regarding Figure 4.5b, the majority of ULDs were stored in KC01 in 2021. This can be attributed to the significant portion of COL and CRT ULDs, as well as significantly long transit times at the hub.



(a) Proportion of SHC on ULD level **(b)** Proportion of ULDs that were stored in KC01

Figure 4.5: Truck-aircraft transit flow analysis

Indeed, it has been concluded that most ULDs have a significantly long transit time, which can be seen in Figure 4.6. In fact, roughly only 20% of all the recorded ULDs in the data set has a transit time of 8 hours or less. Therefore, long transit times of freight have been recognised as a significant contribution to the overall capacity constraints of KC01. However, the reduction of transit times has been considered out of scope for the given research project.

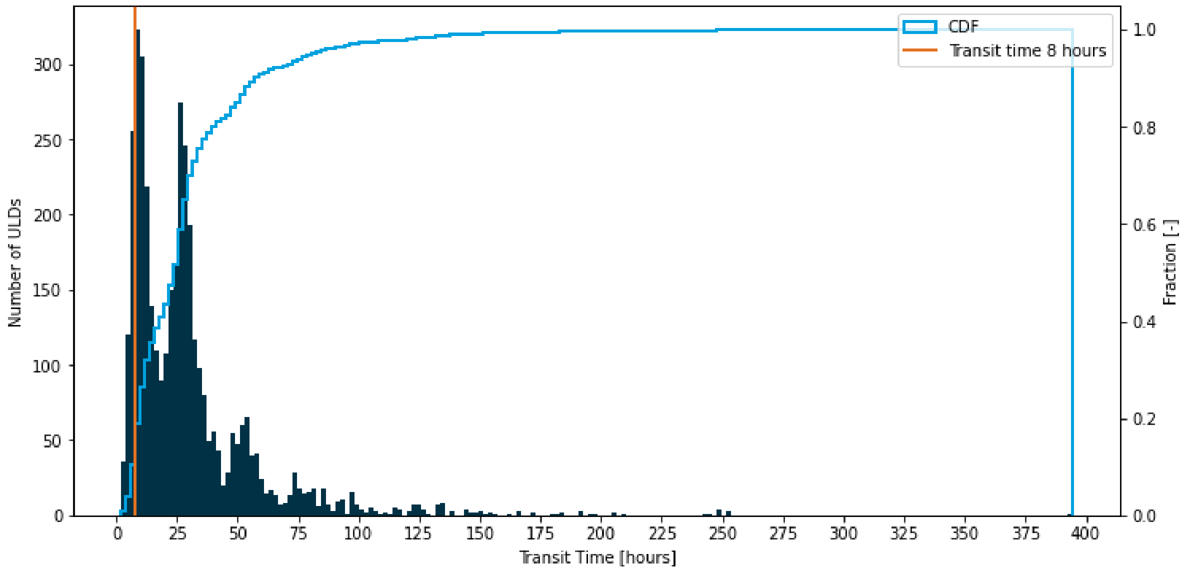


Figure 4.6: Histogram and cumulative distribution function of ULD transit times

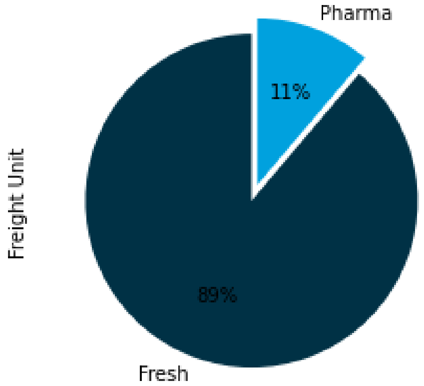
KC01 storage capacity

As mentioned in Section 3.3, cool storage KC01 has a total capacity of 42 COL and 33 CRT and ERT ULDs. Since KC01 is not solely used for pharmaceutical shipments, the storage capacities have been adjusted in order to reflect the research scope. Since the filtered cool chain data set only contains pharmaceutical shipments, the unfiltered data set has been used in order to derive the capacities. In principle, a distinction has been made between fresh and pharma shipments as well as the respective flows at the hub. The resulting shipment type and flow splits have been visualised in Figure 4.7, where the relevant type and flow for the research scope have been indicated with a light blue colour. It can be observed that over 2021, 11% of all COL T-ULDs contained pharmaceuticals, while for CRT all T-ULDs were registered as pharma. Furthermore, for both COL and CRT T-ULDs, the majority of T-ULDs belonged to the studied truck to flight flow at 87% and 84% respectively.

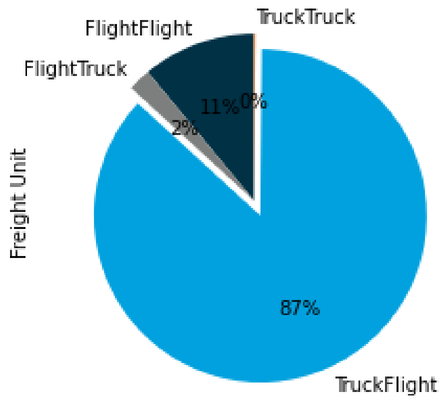
The capacity for each temperature zone in KC01 has been adjusted through the multiplication of the total capacities by the obtained percentages of the relevant shipment types and flows. Therefore, the resulting effective capacity of the COL storage room is $42 \times 0.11 \times 0.87 \approx 4$ ULDs. Similarly, the effective capacity of the CRT storage room is $33 \times 1 \times 0.84 \approx 28$ ULDs. After consultation with the cool chain manager at KLM Cargo, it has been decided to consider on average a slightly higher capacity for the COL storage. This is due to two reasons;

1. Part of the fresh ULDs is handled and stored at a different facility at Schiphol.
2. Due to the capacity constraint regarding KC01, pharma ULDs occasionally receive priority over fresh ULDs in the COL storage.

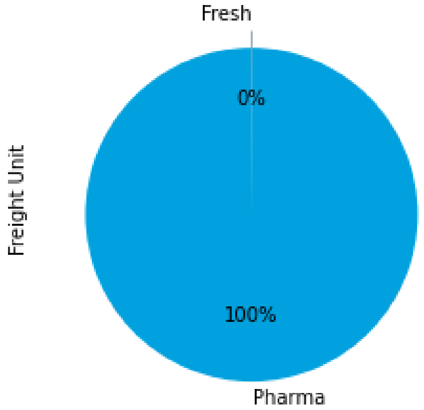
Therefore, the effective capacity of the COL storage room in KC01 has been assumed at 6 ULDs while the effective capacity of the CRT storage room has been kept at 28 ULDs.



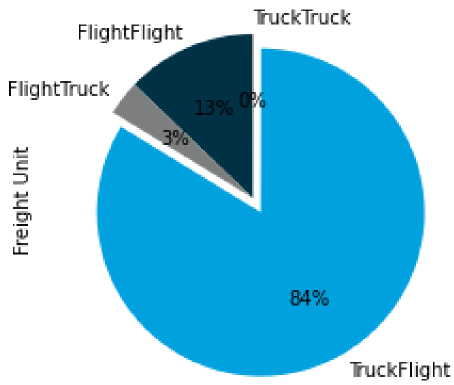
(a) COL shipment type



(b) COL shipment flow



(c) CRT shipment type



(d) CRT shipment flow

Figure 4.7: T-ULD shipment type and flow over 2021 for COL and CRT respectively

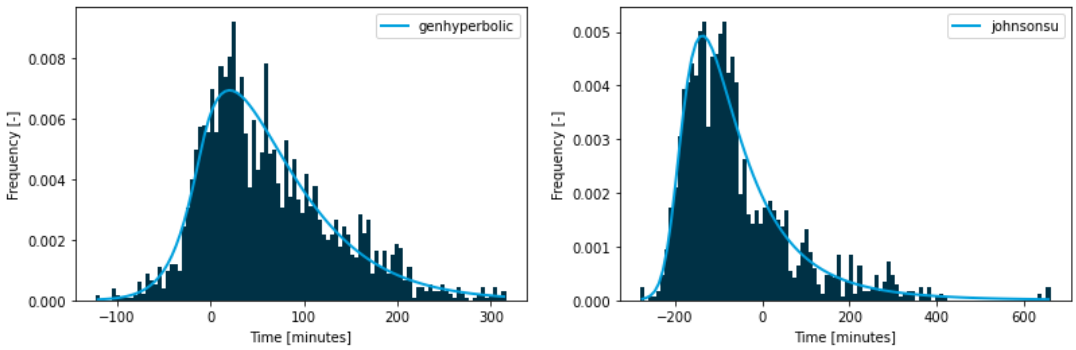
Processing times distributions

The primary use of the analysis data set was the extraction of statistical distributions for the various processing steps in the case study cool chain. The derived distributions serve to represent the variability in the different processing steps in order to properly represent the real-world system. The required statistical distributions have been obtained through the Python fitter package, where the most applicable distribution has been selected based on the minimum squared errors. By means of the statistical distributions, processing times can be drawn in the DES model according to the observed data. For each required processing step, the fitted distributions and respective parameters have been summarised in Table 4.1.

Process	**Distribution**	**Parameters**				
Arrival	Gompertz	$c = 14.29$	loc $= 5.96$	scale $= 432.81$		
PCHS removal	Johnson's SU	$a = -2.14$	$b = 1.44$	loc $= -217.37$	scale $= 60.18$	
PCHS in to KC01 in	Johnon's SB	$a = 1.11$	$b = 1.04$	loc $= 12.35$	scale $= 351.41$	
Transport KC01	Cauchy	loc $= 2.50$	scale $= 0.90$			
KC01 removal	Generalised Hyperbolic	$p = 2.01$	$a = 0.00035$	$b = 0.00018$	loc $= -5.03$	scale $= 0.0086$
PCHS to airside lane	Lomax	$c = 9.63$	loc $= 1.00$	scale $= 39.26$		
Airside lane	Inverse Weibull	$c = 1.38$	loc $= -3.77$	scale $= 14.55$		
Ramp ride	Generalised Hyperbolic	$p = -1.47$	$a = 1.12$	$b = 1.10$	loc $= 10.57$	scale $= 5.60$

Table 4.1: Derived process distributions

In contrast to the other processes, the distributions which have been derived for PCHS and KC01 removal do not represent the average time a ULD spends in either storage facility. Instead, the derived distributions, which have been visualised in Figure 4.8, represent the difference between the removal time rules and the actual time of removal. As mentioned in Section 3.3, the standard removal times from the PCHS and KC01 are 5 and three 3 before STD respectively. Consequently, the distributions shown in Figure 4.8 indicate the variance from the standard removal times as seen in the cool chain data set. As an example, for both Figure 4.8a and Figure 4.8b, 0 minutes indicates the frequency of ULDs removed at the standard removal time. Besides, a time difference of +60 minutes indicates that a ULD has been removed from storage an hour in advance of the standard removal time. Finally, a time difference of −60 minutes indicates that a ULD has been removed an hour later than the standard time of removal. Therefore, regarding Figure 4.8a, most ULDs are typically removed earlier than or before the standard 3 hours for KC01. Regarding Figure 4.8b, the opposite has been observed where ULDs are usually removed later than the standard 5 hours before STD for the PCHS.



(a) KC01 removal time difference distribution

(b) PCHS removal time difference distribution

Figure 4.8: Fitted distributions for the storage removal times

Current performance

From the cool chain data set, the current performance of the KLM Cargo cool chain has been determined on the basis of seven KPIs, as shown in Table 4.2. The current performance on these KPIs have been considered as the input for the validation of the PT model, which has been further discussed in Section 4.2. Each deadline has been computed on ULD level. Furthermore, for ULDs which have not been stored in KC01, the TOR has been calculated similarly and thus represents the total time spent at the hub. Regarding cool chain deadline 3 in Table 4.2, the data matches the observed pattern in Figure 4.8a; the majority of ULDs are removed more than 3 hours before STD, which is also indicated by a low score of 17.23% on this deadline.

	Cool chain deadlines			**Handling deadlines**			**TOR** [%]
	1 [%]	**2** [%]	**3** [%]	**1** [%]	**2** [%]	**3** [%]	
Truck-aircraft pharma	96.86	61.86	17.23	92.74	92.28	84.02	52.02

Table 4.2: Current performance based on cool chain data set

4.2. Physical Twin

The presented research in this report has been accomplished by manually programming a DES model in Python 3.9 compiled in the scientific Python development environment Spyder, by use of the salabim package for Python. The programming code can be observed in Appendix B.1. The model development and implementation have been performed on a Lenovo ThinkPad L490 with an Intel(R) Core(TM) i5-8365U CPU running at 1.60 GHz using a 64-bit architecture and 16.0 GB of RAM. Following the DT development approach introduced by Ait-Alla et al. [23] and discussed in Section 2.6, the DES model which has been developed provides a virtual representation of the studied system: the PT. The primary aim of the so-called PT has thus been to replicate the physical system behaviour. The replication has been achieved by the programming of the system and business logic while capturing the system variance with the presented distributions. In order to measure the performance of the replicated physical system, the proposed OCCE performance evaluation with the applied KPIs has first been discussed. The PT has been further elaborated on by first considering a conceptual overview of the model, after which the simulation objects have been further discussed. Since any model is a simplification and abstraction of reality, the assumptions which have been made for the development of the model have been discussed in order to finalise the section. However, firstly a nomenclature has been provided with the variables that have been used for the remainder of the paper.

Nomenclature

C_{COL} Total capacity of the KC01 COL storage room

C_{CRT} Total capacity of the KC01 CRT storage room

F Set of multiplication factors for the sensitivity analysis

I The set of availability measurements

n Number of simulation runs

q_{col} Current quantity of stored ULDs in the KC01 COL storage room

q_{crt} Current quantity of stored ULDs in the KC01 CRT storage room

R_A Average cool storage availability of the KC01 storage facility

$R_{A,COL}$ Cool storage availability rate of the KC01 COL storage room

$R_{A,CRT}$ Cool storage availability rate of the KC01 CRT storage room

R_{KC01} Standard KC01 storage removal time rule

R_{OTP} On time performance rate

R_{PCHS} Standard PCHS storage removal time rule

R_{TA} Temperature adherence rate

R_{TT} Cool storage transit time business rule

S Total quantity of handled AWBs

s_e Number of AWBs with all ULDs having an exposure less than eight hours

s_{mf} Quantity of AWBs with a ULD that missed the flight

4.2.1. Performance evaluation

The OCCE framework which has been proposed in Section 2.4 has formed the basis for the performance evaluation of the developed model and also serves as the metric to provide a quantitative answer to the main research question. Since the metric has been comprised of three rates, each rate has been further discussed separately.

Cool storage availability

The cool storage availability rate provides an insight into the availability of cool storage infrastructure with regard to the respective capacity which is typically constrained. The availability has been considered as the ratio of residual capacity and total capacity for the given cool storage area, or more formally:

$$R_{A,COL} = \frac{C_{COL} - q_{col}}{C_{COL}} \tag{4.1a}$$

$$R_{A,CRT} = \frac{C_{CRT} - q_{crt}}{C_{CRT}} \tag{4.1b}$$

$$R_A = \frac{\frac{\sum_{i=0}^{I} R_{A,COL}}{|I|} + \frac{\sum_{i=0}^{I} R_{A,CRT}}{|I|}}{2} \tag{4.1c}$$

The availability rates in Equation 4.1a and Equation 4.1b provide only the cool storage availability of the respective room at the time of measurement. The availability of either storage room can thus be used in the presented form for cool storage decision-making. However, for model output derivation, an average availability score over the total simulation run time is taken. Consequently, the average of $R_{A,COL}$ and $R_{A,CRT}$ is taken in order to obtain the average cool storage availability shown in Equation 4.1c.

On time performance

The timeliness of a cool chain may be represented by the on time performance rate. In the presented case study, the timeliness of the cool chain process has been represented through the FAP KPI. The FAP indicator is measured on shipment AWB level, which consequently means that an AWB has not been flown on time if any of its corresponding ULDs have missed the flight. Therefore, the on-time performance can be measured as the fraction of AWBs which have ULDs that missed the corresponding flight connection and the total amount of handled AWB, or more formally:

$$R_{OTP} = 1 - \frac{s_{mf}}{S} \tag{4.2}$$

Temperature adherence

The temperature adherence rate represents the accordance to the required environmental conditions of freight in a cool chain. Although KLM Cargo currently measures the TOR, the KPI has not been considered adequate since the actual environmental conditions have not been considered apart from the time spent in cool storage. Therefore, it has been decided to consider the exposure as a measure for temperature adherence, where the exposure has been defined as the total time a ULD has spent outside of the required temperature range. The notion of exposure has been visualised in Figure 4.9. In principle, any temperature within the specified temperature range of the pharma freight is acceptable and does not contribute to harmful exposure. In other words, exposure of freight is recorded as soon as the ambient temperature surrounding the ULD is outside of the Lower Temperature Bound (LTB) or Upper Temperature Bound (UTB). In this way, the exposure can be calculated as the total time between recorded timestamps where the temperature was above the UTB or below the LTB. Currently, KLM Cargo does not possess ULDs with integrated temperature sensors. Therefore, the ATAL warehouse sensors and ambient temperature database have been used as data sources. Although the accuracy of these sources is limited compared to integrated devices, the method of quantifying temperature adherence is the same. It has also been noted that realistically, temperature changes would happen less discretely, as indicated by the striped lines in Figure 4.9. Therefore, the time at which the temperature has exceeded either the UTB or the LTB would be earlier than the timestamp recorded in the developed model. Therefore, it has been acknowledged that the accuracy in this respect could be improved by either integrated temperature sensors or an increase in temperature logging frequency.

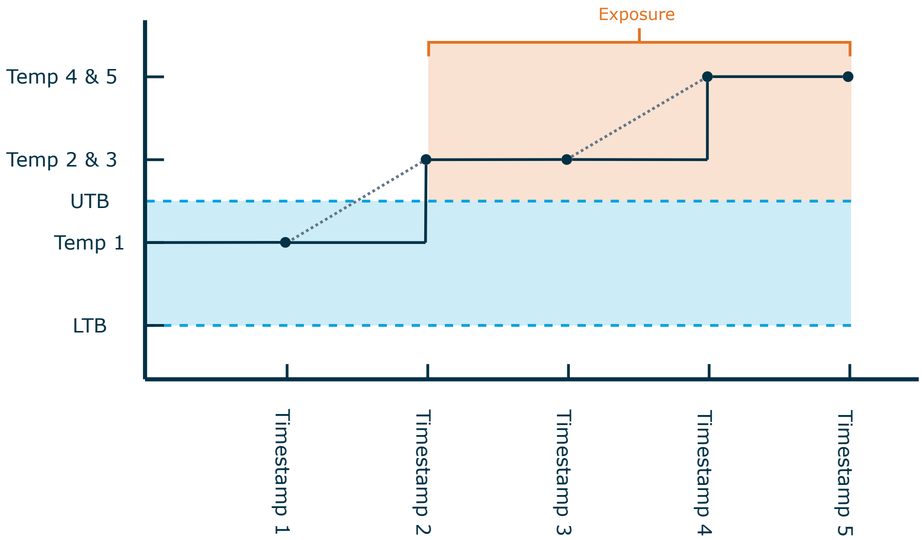


Figure 4.9: A visualisation of the notion of exposure in a cool chain

Similarly to the on time performance, the temperature adherence has been measured on AWB level. At the hub, a maximum exposure of 8 hours has been deemed acceptable and is agreed upon by shippers. The maximum exposure of 8 hours also acts as the reasoning behind the cool storage rule with a transit time of more than 8 hours. Therefore, an AWB has an acceptable exposure if the exposure of all corresponding ULDs is below 8 hours, resulting in the following formal definition:

$$R_{TA} = \frac{s_e}{S} \tag{4.3}$$

OCCE

Each rate which has been described previously has been combined into the overarching performance metric for the studied cool chain; the OCCE which has been visualised in Figure 4.10. In principle, the proposed performance metric for a cool chain provides a flexible method for capturing the performance of a cool chain as a whole. For the presented case study, three lower level KPIs have been used to represent the three rates of the OCCE, while providing an overall performance score.

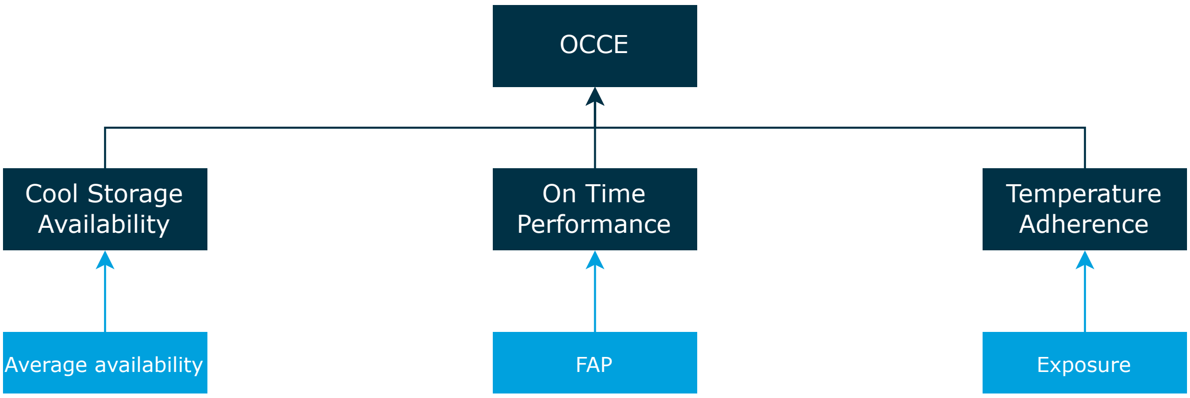


Figure 4.10: The OCCE metric scheme for model performance evaluation

The presented scheme in Equation 2.2 has been used as model output for performance evaluation. As discussed in Section 2.4, the OCCE metric is obtained through the multiplication of the three underlying rates:

$$OCCE = R_A \times R_{OTP} \times R_{TA} \tag{4.4}$$

4.2.2. Conceptual model

The proposed conceptual research model has been shown in Figure 4.11. In line with the proposed DT definition and development methodology discussed in Section 2.6, the proposed system has been characterised through the physical system and digital system. The physical system contains the real system, i.e. cool chain, which is situated in a given environment. Furthermore, decision-makers are actively interacting with the physical system with regards to cool storage decisions and ULD transportation. Although a DT has been characterised by an automatic data flow between the digital and physical systems, the research has been limited to the digital system; more specifically the PT and DT. The connections between the physical and digital systems have thus been represented by a dashed arrow. The data sets which have been described in Section 4.1, have been indicated in Figure 4.11 and form the input of the PT; the DES model. The DT has consequently been proposed as the part of the digital system which offers additional functionality to the PT in the form of a decision support module. It should be noted that the decision support module has been further discussed in Chapter 6. In this chapter, the development of the physical twin has been described, along with the verification and validation of this simulation model in Chapter 5.

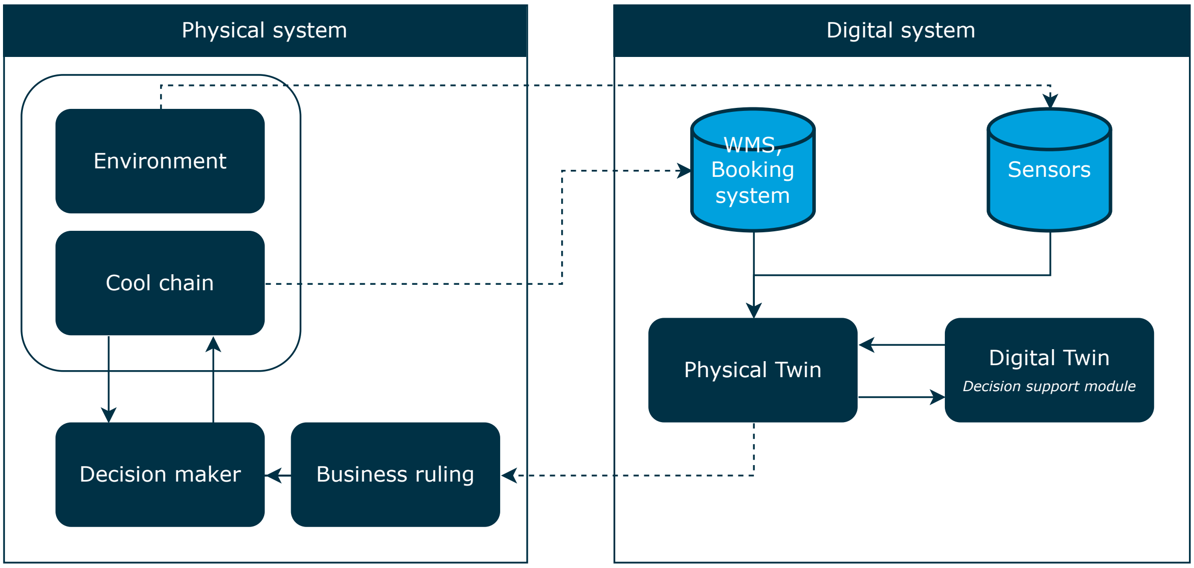


Figure 4.11: Conceptual research model

From Figure 4.11, a more in-depth overview of the digital system has been shown in Figure 4.12, where an emphasis has been placed on the PT model for the current chapter. Given the research scope, the input of the PT model is historical data input files on ULD arrivals as well as warehouse and ambient temperatures as described in Section 4.1. The resulting output of the simulation model includes the individual OCCE rates as well as the overall metric which can be calculated from it.

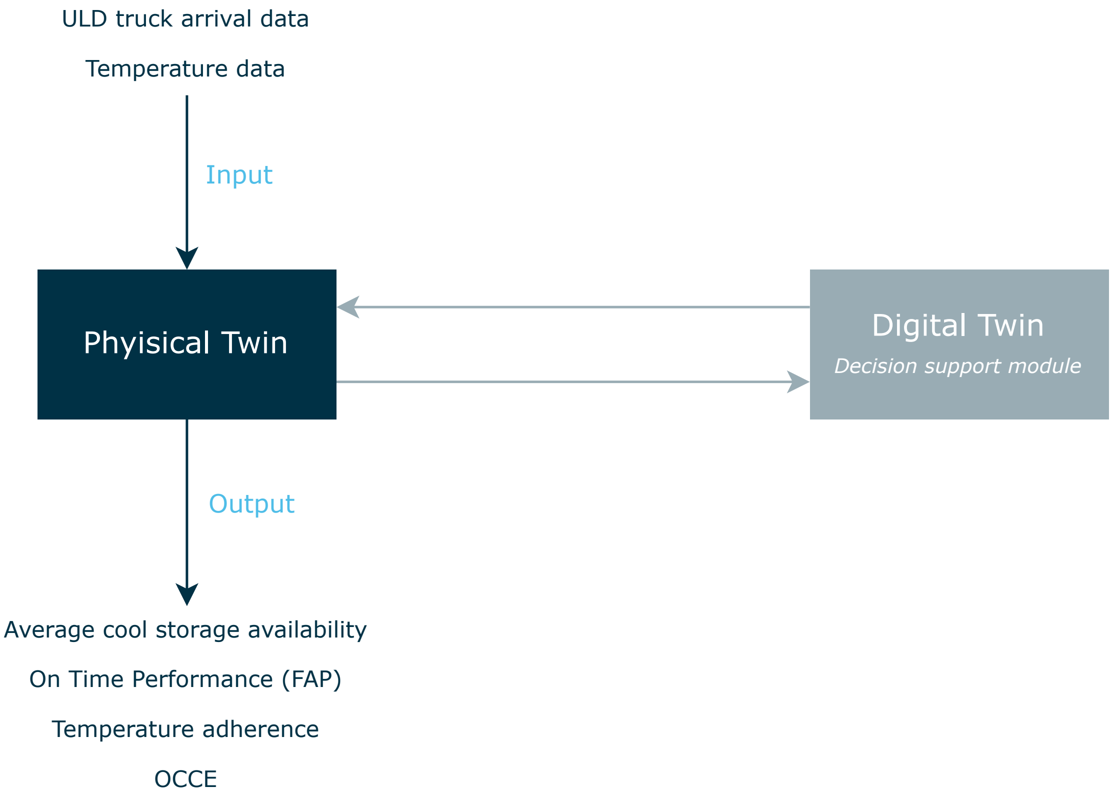


Figure 4.12: The PT model in the digital system

In the current situation reflected by the PT, the decision-making with regards to storage of ULDs in KC01 is done solely on the basis of the introduced business ruling described in Section 3.2. The current decision-making process based on the business rules has been visualised in Figure 4.13. Depending on the product code of a T-ULD and thus its corresponding SHC, the static business rules are applied which determine whether a T-ULD is placed in cool storage or not. Therefore, the input data, as indicated by the striped arrows, include primarily the SHCs of each individual T-ULD. Besides, in order to verify whether the storage is full, the previously assigned ULDs which are still in transit towards the storage facility should be taken into account.

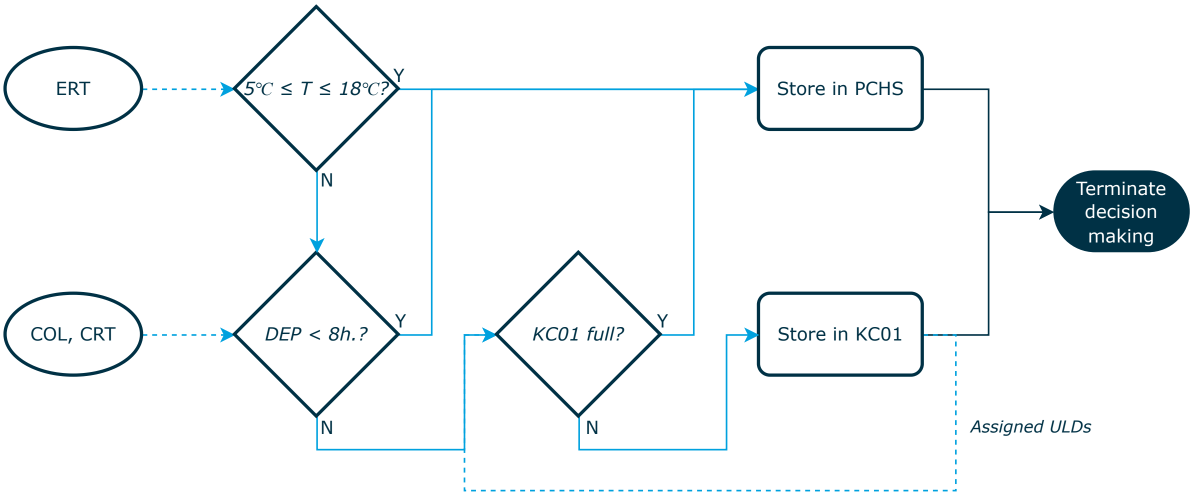


Figure 4.13: Current cool storage decision-making flow chart based on static business ruling

From the digital system shown in Figure 4.12, a more in-depth visualisation of the PT has been shown in Figure 4.14. The overview provides a flow chart of the PT process and thus the activities which are performed in the DES model. The actual entities which flow through the model are the T-ULDs, which arrive into the cool chain according to the input data set. Depending on the characteristics of the freight and thus the T-ULD, the freight unit follows a certain path throughout the cool chain which depends mainly on whether it is a late arrival or receives cool storage or not. Furthermore, from the temperature data set, ambient outside, as well as warehouse temperature, has been extracted, which is pushed hourly to the relevant locations in the process. As an example, the outside temperature is updated hourly according to the historical input data. The so-called building blocks of the DES model have been further discussed in Section 4.2.3.

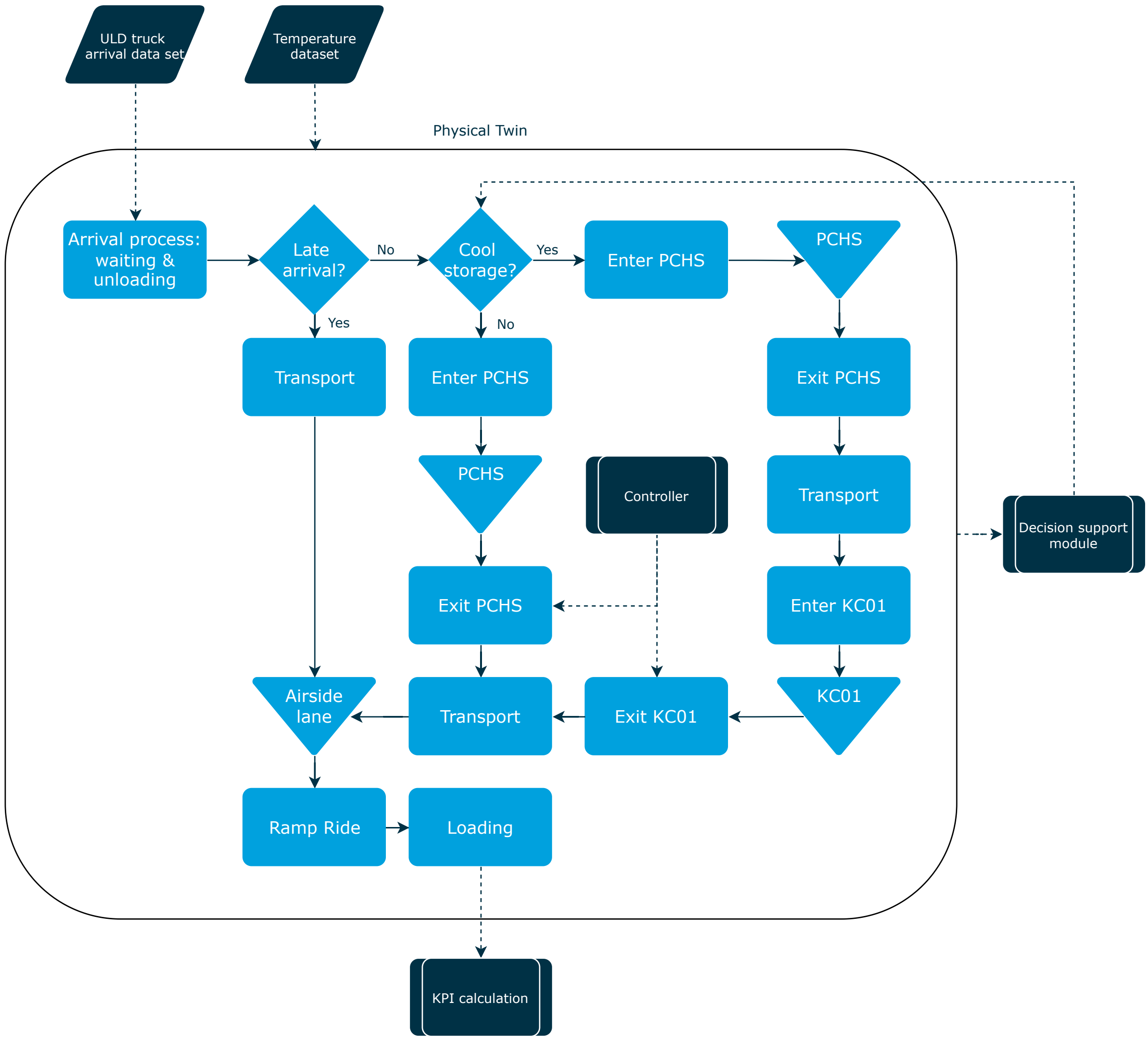


Figure 4.14: PT model flow chart

4.2.3. Simulation objects

In Table 4.3, a description and main characteristics have been summarised of each simulation object which has been programmed in order to develop the DES physical twin. By means of the salabim package, a DES model can be programmed using objects and resources. Objects may either be active or passive, where an active object contains a pre-defined process and a passive object can be seen as a data component. Although technically, the transporters and tractors are resources used in the cool chain process, these components have been modelled as active objects for increased programming flexibility. The InputGenerator and TempGenerator objects can be considered as a special type of object; a generator. Both generators create the input of the DES model according to the input data set files. The remainder of objects have been programmed in order to provide the correct interaction with other objects such that the ULDs flow through the process according to the actual physical system behaviour. Furthermore, the storage spaces such as KC01 and the PCHS have been modelled as queues in which the FreightUnits are placed and consequently removed. In order to regulate the cool storage and execute storage retrieval in general, two objects have been programmed. Firstly, the decision support module has been programmed as a distinct object which decides whether ULDs are placed in cool storage or not. For the PT, which replicates the actual physical system, the decision-making process has been modelled according to Figure 4.13. Consequently, this object can be altered according to function as the proposed decision support module, or DT. Finally, the controller object functions as a simplified WMS which orders ULDs to be removed from either the PCHS or KC01 storage facility. Similarly to the decision module, in the current situation, the controller utilises the standard removal rules in combination with the derived distributions for the time of removal of ULDs.

Real entity	Simulation object	Description	Main characteristics
-	InputGenerator	Generates ULDs according to the input data set	Input data set
-	TempGenerator	Generates and updates the hourly temperatures according to the temperature data set	Temperature data set
AWB	AirWaybill	Is linked to the corresponding ULDs for temperature adherence and OTP calculations	Corresponding ULDs
ULD	FreightUnit	Entity which is generated, flows through the system and is finally terminated	STD, transit time, SHC, temperature range
WMS	Controller	Assigns removal time from storage to FreightUnit	Removal time rules, removal time distributions
Truck	TruckArrival	Imitate the arrival and unloading of FreightUnit	Arrival and unloading distribution
-	DecisionModule	Determine whether FreightUnit is stored in KC01 according to the business rules	Cool storage business rules
PCHS	PCHSEntrance	Accepts entering FreightUnit and places it in storage	PCHS storage, temperature zones, processing time
	PCHSExit	Removes FreightUnit from storage	PCHS storage, temperature zones, processing time
KC01	KC01	Provides cool storage for FreightUnit according to SHC	COL storage, CRT storage, ETV processing time, temperature set points
Transporter	Transporter	Transportation of a single ULD to the airside lanes and KC01	Transport time distribution
Tractor	Tractor	Transportation of late arrivals to airside lanes and ramp rides to the aircraft	Ride duration, ramp ride distribution
Air side lanes	Air side lanes	Buffer lanes on air side in front of the warehouse	Queuing time distribution

Table 4.3: Description and main characteristics of the DES model objects

4.2.4. Model assumptions

In order to develop a DES model within an acceptable time frame for this research project, several assumptions have had to be made. The assumptions have been further elaborated and classified into general assumptions and assumptions related to the infrastructure and equipment for the presented case study.

General

1. With regard to the environmental conditions, only temperature has been taken into account in this study. Although humidity and light exposure may especially affect freight outside, it has not been regarded for the sake of modelling simplification.
2. The temperature exposure has been assumed to be a cumulative time during which the temperature surrounding the freight was above the respective UTB or below the LTB.
3. T-ULDs which are classified as late arrivals are not entered into the PCHS and are instead transported towards the airside lanes.
4. Although there are a significant number of factors influencing the FAP KPI, any external factors have not been taken into account. Therefore, the FAP metric only captures ULDs which did not make the connecting flight due to excessive processing times as obtained from the model processing time distributions.

Infrastructure and equipment

1. It has been assumed that there are two transporters available for the transportation of T-ULDs from the PCHS exit to the airside lanes.
2. The total number of available tractors in the system has been assumed at 5 units. Although a larger quantity of tractors is available each day in the physical system, the total number has been reduced in order to reflect the maximum pulling capacity of five dollies which are also used by other freight types and configurations.
3. It has been assumed that the ramp ride time of a tractor back towards the warehouse is equal to the time it took to get to the aircraft.
4. It has been assumed that it takes five minutes for late arrival T-ULDs to be transported to the airside lanes by a tractor.
5. It has been assumed that the ULD removal rate of the ETV in KC01 is equal to six per hour, or ten minutes per ULD.
6. The temperature set points of KC01 have been assumed to be perfectly constant. In other words, it has been assumed that the cool storage facility has a reliability of 100%.
7. It has been assumed in accordance with the physical system that outbound ULDs from KC01 always have priority over inbound ULDs into KC01.
8. The respective input temperature for the entrance, storage and exit parts of the PCHS has been assumed to be uniform over each respective part. Therefore, for example, it has been assumed that the input temperature data for the PCHS storage area is uniform over the whole storage area. In the physical system, the temperature has known to be varying from different areas even within the same part of the system. Nonetheless, this has not been taken into account in the case study.

4.3. Model development synthesis

In this chapter, the required modelling for the improvement methodology has been described and constructed by first considering the necessary data collection, handling and analysis of the studied system. It has additionally been noted that during this stage, the data quality has been found to be poor. Then, in order to obtain a virtualisation of the physical system, the PT model has been described through the introduction of the implemented OCCE metric, the conceptual improvement model as well as the objects from which the DES model has been built. However, before the implementation of the DT concept and thus the complete implementation of the digital system, it has been deemed necessary to prove whether the developed PT model is indeed correct and can be used for the completion of the research. Therefore, in the following chapter, model verification and validation have been performed.

5

Verification and validation

As part of model development, in the following chapter, a discussion has first been provided on a convergence analysis of the simulation model output. Consequently, the verification and validation of the DES model has been discussed in Section 5.2. Furthermore, in order to study the sensitivity of the simulation model to data input and process parameters, a sensitivity analysis has been performed and discussed in Section 5.3. Finally, an answer has been constructed to the following research question:

4 To what extent does the developed model effectively represent the research object, in terms of verification and validation??

In order to guide and clarify the structure of the following chapter, an experimental research plan for the verification and validation has been provided in Figure 5.1.

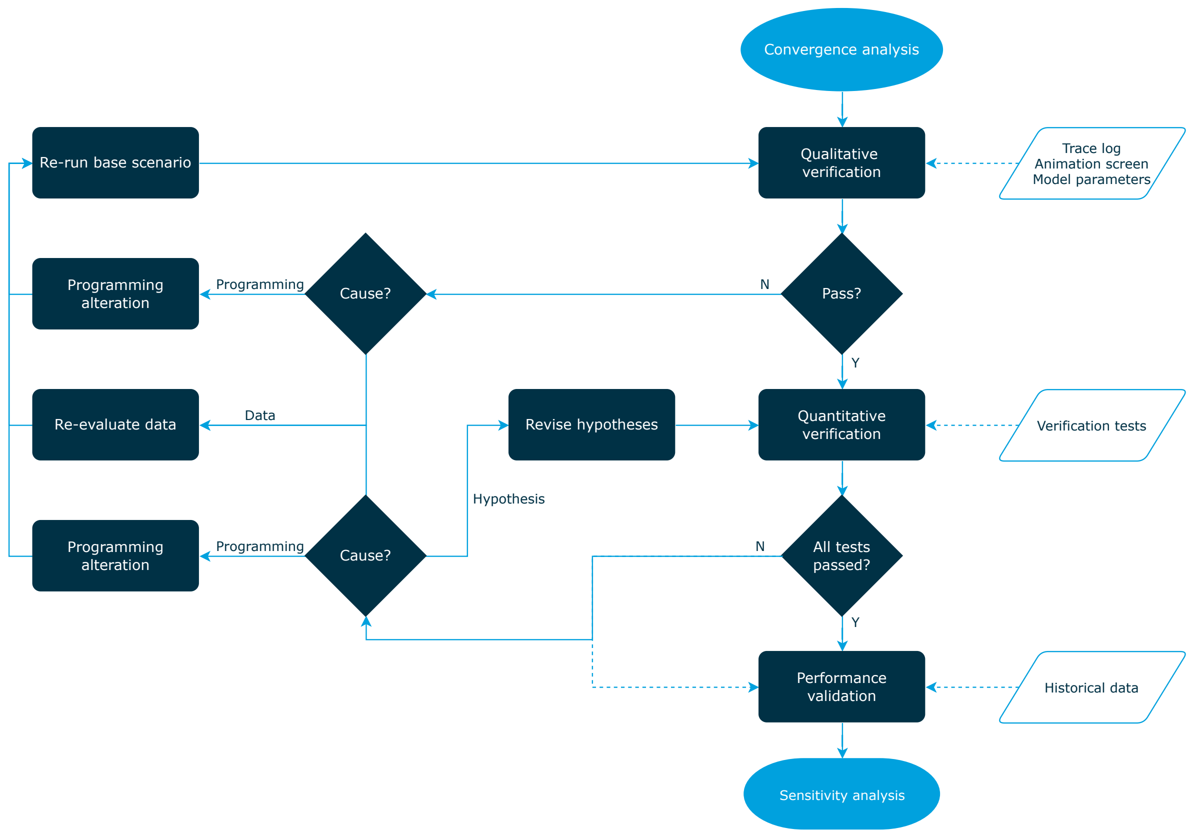


Figure 5.1: Experimental research plan for model verification and validation

As seen in Figure 5.1, the initial step which has been performed is a convergence analysis with the goal of finding the required number of simulation runs n in order to account for the stochastic nature of the DES model. Then, several iterations of the research plan have been performed consisting of qualitative verification, followed by quantitative verification and validation. The qualitative verification has been performed using the trace log, animation screen and model parameters. For the quantitative verification, several tests with corresponding hypotheses have been defined by means of changing model parameters. In the case that either verification step was not successful, alterations have either been made in terms of programming, input data or the specified verification test hypotheses. Finally, the model validation has been performed using historical data. The experimental plan has been finalised with a sensitivity analysis. The aim of the experimental research plan for verification and validation was to reveal any model errors as well as validate the model performance. Therefore, referring to Figure 5.1, in the case that the verification tests have not been passed, validation results have been collected nonetheless in order to specify the model validation during that specific iteration. Consequently, it can be studied whether any changes made to the model had an influence on the model validation as well. Each respective aspect and iteration of the experimental plan has been further elaborated in the following sections.

5.1. Convergence analysis

In order to study the convergence of the PT model output, i.e. the OCCE and the individual rates, the number of simulation runs n has been increased while averaging the resulting KPI rates. In principle, the convergence analysis has been used in order to study the convergence of the model while taking into account the computational expenses of repeating simulation runs for a total of n times. The resulting model convergence has been plotted and is shown in Figure 5.2. For an increasing number of runs until $n = 30$, the rates and corresponding OCCE have been plotted along with the $\pm 1\%$ limits of each KPI at $n = 30$. In principle, it has been expected that the variation in KPI values should decrease with an increase of n. However, an increase in simulation runs comes at the expense of increased computational time. A single simulation run has been recorded at roughly 90 seconds. Therefore, for each additional run, the computational time is extended by one and a half minutes. For the convergence analysis, this has been found to be a limiting factor for the maximum number of n. For $n = 30$, the computational time has been estimated at $\frac{\sum_1^{30} n \times 90}{3600} \approx 11.5$ hours. The computational time necessary for an incremental number of simulation repetitions n has been shown in Figure 5.3.

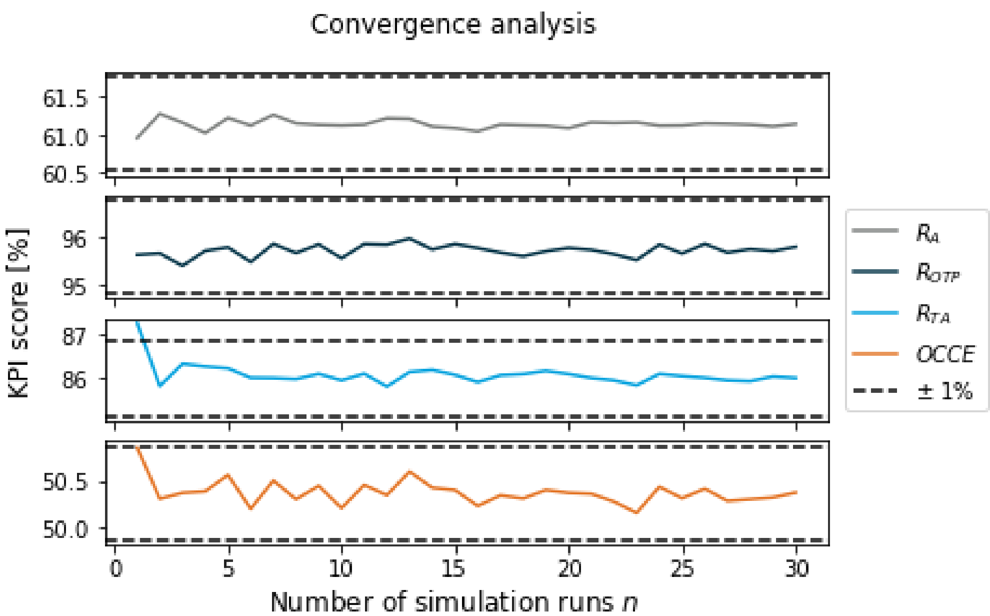


Figure 5.2: Physical Twin model convergence analysis for a total of $n = 30$ runs

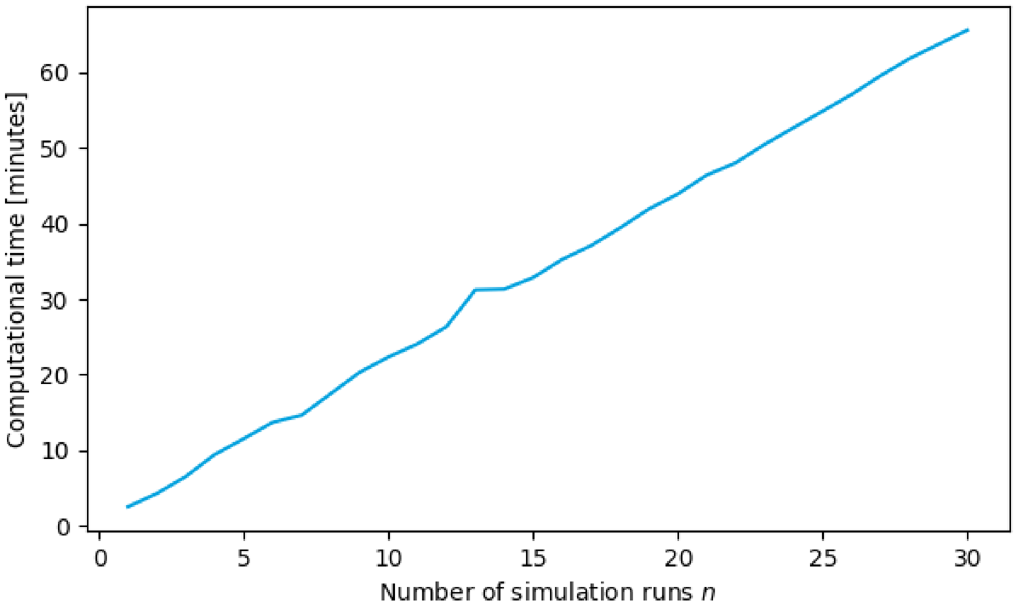


Figure 5.3: Computational time for n repeated simulation runs

From Figure 5.2, several conclusions have been drawn. Firstly, it has been observed that the variability of the individual OCCE rates R_A, R_{OTP} and R_{TA} is limited. As an example, it can be seen that the value of R_A remains within 61.0% and 61.5%, well within the ±1% range of the value at $n = 30$. The latter has also been found for R_{OTP}. Similarly, the variations in R_{TA} swiftly remain within the ±1% range. Since the OCCE metric is the result of a multiplication of the three rates, the value is more sensitive to variations. Nonetheless, the OCCE value remains well within the ±1% range at $n = 30$ while the percentile fluctuations have been found to be limited. Based on the convergence of the three rates as well as the OCCE metric and the computational time, it has been concluded that $n = 20$ simulation runs are sufficient. At $n = 20$ repetitions the variation has been limited while the computational time has been deemed acceptable at under an hour.

5.2. Model verification and validation

As shown in Figure 5.1, following the convergence analysis, several iterations of model verification and validation have been performed. For clarity, a definition from the literature has first been discussed for verification and validation respectively. A formal definition of model verification has been stated as *"ensuring that the computer program of the computerised model with its solution method and the computer program's implementation are correct"* [119]. In other words, model verification is aimed at answering the following question: *is the model right?*. Right in this sense has been understood as the degree to which the model implementation meets the specification. Verification can be approached both qualitatively by manually examining the behaviour of the model or quantitatively by examining the model output against specified expectations. Similar to model verification, Sargent [119] has stated the formal definition of model validation as the *"substantiation that a computerised model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model"*. Therefore, the process of model validation has been aimed at answering the question; *is it the right model?*. Effectively, a validated model should specify whether the developed model is an accurate representation of the real system. It should be noted that, although qualitative verification has been performed each iteration, it has not been discussed after the first iteration. This is due to the fact that each time the same approach has been used as in the first iteration and only a failure of qualitative verification would have been worth mentioning at this point. Furthermore, despite the fact that the validation results have been collected each iteration, it has only been shown if any changes have been observed after subsequent iterations For clarity, the respective methods of qualitative and quantitative verification and validation have been further described below. Afterwards, with a clear definition and approach of verification and validation, each respective iteration of the experimental plan has been further discussed.

Qualitative verification

Besides during model verification, qualitative verification has been thoroughly used throughout the development of the DES model in order to expose any logic and programming errors early on in the development process. The primary qualitative verification method which has been used is by means of event tracing. The salabim package offers the possibility to extract a trace log from a simulation run in which events can be printed. The trace log has then been used to check event order, causal relations and event times in order to examine whether the simulated process is in line with the physical system counterpart. A part of the trace log has been shown in Figure 5.4.

```
      Fri 2021-01-01 05:17:00 truckarrival.0       current
                                                   PMC73557                         leave Shipment Arrival Queue
                                                   PMC73557 hold +00:17:28          scheduled for Fri 2021-01-01 05:34:28
                                                   truckarrival.0 passivate
      Fri 2021-01-01 05:34:28 PMC73557             current
                                                   PMC73557                         enter Apply Business Ruling queue
                                                   decisionmodule.0 activate        scheduled for Fri 2021-01-01 05:34:28
                                                   PMC73557 passivate
      Fri 2021-01-01 05:34:28 decisionmodule.0     current
                                                   COL storage full set             value = False
                                                   CRT storage full set             value = False
                                                   PMC73557                         leave Apply Business Ruling queue
                                                   PMC73557                         enter ULDs assigned to CRT storage
                                                   PMC73557 activate                scheduled for Fri 2021-01-01 05:34:28
                                                   decisionmodule.0 passivate
      Fri 2021-01-01 05:34:28 PMC73557             current
ERT   **INFO**                PMC73557             CoolStorage:                     Yes
                                                   PMC73557                         enter PCHS Entrance Queue
                                                   pchsentrance.0 activate          scheduled for Fri 2021-01-01 05:34:28
                                                   PMC73557 passivate
      Fri 2021-01-01 05:34:28 pchsentrance.0       current
                                                   PMC73557                         leave PCHS Entrance Queue
                                                   pchsentrance.0 hold +00:15:00    scheduled for Fri 2021-01-01 05:49:28
```

Figure 5.4: Simulation trace log output

From left to right, the trace log has been built up with three columns. The first column shows the timestamp at which the DES model performs an event including for which model object. The second column shows either a simulation object which is currently active or it shows an activity assigned to that object. Finally, the third column indicates either an action such as leaving a process queue or it shows when a current object is scheduled for a future event. As an example, starting from the top, it can be seen that on 2021-01-01 at 05:17:00 a truck arrived according to the input data set, which contains the T-ULD with serial number PMC73557. The truck and thus T-ULD wait according to a sample drawn from the arrival and loading distribution, after which the T-ULD is scheduled as the next occurring event in this case. In a similar manner, the trace log can be used to trace individual ULDs while recording all process events and timestamps. Furthermore, an ***INFO*** line has been programmed indicating the SHC of the T-ULD, the serial number and the decision whether it is placed in cool storage or not. Such information has aided the qualitative verification by means of the simulation trace log.

Similarly to the trace log, salabim provides a built-in method for simulation model animation which can be used to animate and thus examine the model behaviour. In specific, the animation screen aids in verifying whether the T-ULDs logically follow the processing steps and do not take any impossible routes. For reference, the animation screen has been shown in Figure 5.5. The information shown on the animation screen has been manually selected. This information includes the T-ULDs with their corresponding serial numbers and the current position in the cool chain such as the PCHS or KC01, the number of T-ULDs currently in either cool storage rooms and for instance a process metric such as total processing time. The animation screen visualises the DES model as it evolves in time and has thus provided a means to qualitatively verify the model behaviour as it is run.

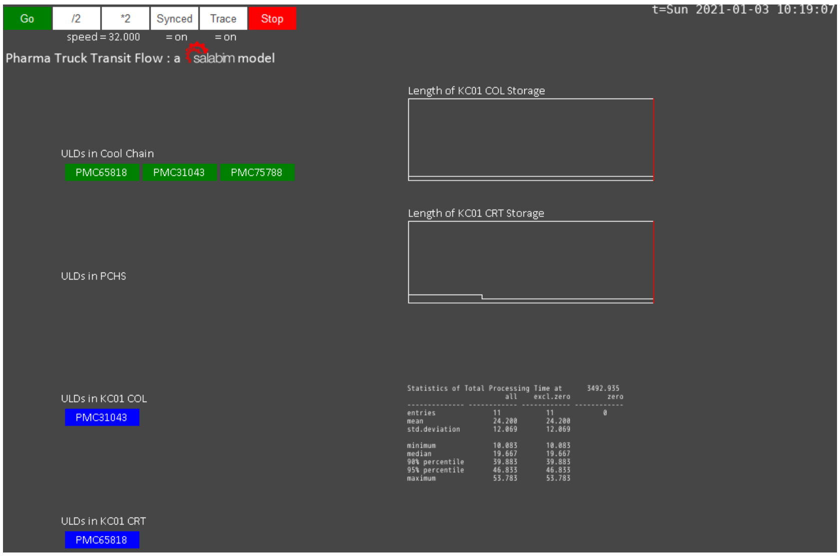


Figure 5.5: Simulation model animation screen

Finally, the model has further been qualitatively verified by considering the model output parameters. As an example, it has been checked that the number of ULDs going into the system is equal to the number of handled ULDs. Consequently, it can be concluded that all ULDs fully flow through the cool chain system in a correct manner as verified through the trace log and animation screen.

Quantitative verification

In order to provide quantitative verification of the PT model, a base model run has been performed with default simulation settings in accordance with the presented case study. To verify whether the model is right and follows expected behaviour, several verification tests have been defined by changing simulation parameters after which certain hypotheses for model behaviour have been summarised. If the model KPIs reflect the expected behaviour in a test, the model can be said to have a Pass on that test. Once a Pass has been scored on each test, it has been assumed that the model has been quantitatively verified. In the case that a Fail has been obtained, it has been attempted to alleviate the issue and improve the model. Logically, the verification and validation in general is thus an iterative process. The base scenario parameters which have been used in the verification tests have been shown in Table 5.1 and the corresponding values reflect the presented case study.

Parameters	**Description**	**Values**
C_{COL}, C_{CRT}	KC01 storage capacities	$C_{COL} = 6$, $C_{CRT} = 28$
R_{TT}	Cool storage transit time business rule	$R_{TT} = 8$ hours
R_{PCHS}, R_{KC01}	Storage removal time rules	$R_{PCHS} = 5$ hours, $R_{KC01} = 3$ hours

Table 5.1: PT model base scenario parameters

Correspondingly to the base parameters shown, the resulting scores of the OCCE metric averaged over $n = 20$ runs have been obtained in order to compare the verification test results with the predefined hypothesis. The results of the base simulation runs have therefore been shown in the hypotheses. It has been noted that in the case that any modifications to the input data or programming have been made, the base scenario has been re-run before the start of a new verification and validation iteration. Each hypothesis has been formed by considering what should happen to the model output KPIs when a specific parameter has been changed. Each test has been further discussed below;

1. **Doubling of capacities**: with a doubling of the capacity of the KC01 cool storage rooms, it has been anticipated that logically the availability rate should increase since the availability of cool storage is directly related to the capacity of the facility. Furthermore, since with a doubling of

the capacity the number of T-ULDs which have been cool stored increases, it has been expected that the on time performance decreases. This is due to the fact that storing a T-ULD in KC01 required additional processing steps with the ETV being a bottleneck in the cool storage facility. Therefore, there is an increased chance that a T-ULD may not be handled on time in order to make the connecting flight. Finally, the temperature adherence rate has been expected to increase since more T-ULDs could be stored in KC01.

2. **Halving of capacities**: it has been expected that compared to the doubling of the capacities, the halving of the capacities has the opposite effect on each OCCE rate following the same logic; a decrease in availability, an increase in on time performance and a decrease of the temperature adherence.
3. **Doubling of cool storage business rule**: for a doubling of the cool storage transit time business rule from 8 hours to 16 hours, it has been anticipated that in principle less T-ULDs are stored in KC01 given the histogram of T-ULD transit times shown in Section 4.1.3. Consequently, it has been expected that the availability rate would increase, the on time performance rate would increase and the temperature adherence would decrease.
4. **Halving of cool storage business rule**: a halving of the cool storage business rule to 4 hours should result in more T-ULDs being stored in KC01 since similarly but opposite to test 3, T-ULDs with a shorter transit time than 8 hours should now also be stored in cool storage. Consequently, a decreased availability and on time performance rate has been anticipated. However, despite the expectation that more T-ULDs receive cool storage, it has been expected that the temperature adherence slightly decreases as well. This is due to the fact that although the storage rule has been reduced to 4 hours, the temperature adherence is still measured against an allowed exposure of 8 hours. Therefore, as an example, a T-ULD with a transit time less than 8 hours may compete with a T-ULD with a transit time longer than 8 hours for storage in KC01.
5. **Halving of standard removal times**: by halving the standard removal time rules to 2.5 and 1.5 hours for the PCHS and KC01 respectively, T-ULDs should remain in storage for longer and are thus subject to a larger chance of missing the connecting flight while experiencing less exposure. Therefore, a decrease in the availability and on time performance rate has been expected while an increase in temperature adherence has been anticipated.
6. **Doubling of standard removal times**: opposite to test 5, a doubling of the removal times to 10 and 6 hours respectively should result in shorter storage times and thus a lower chance of a missed flight and a higher chance of exceeding the allowable exposure. Consequently, the opposite hypothesis has been formulated compared to test 5.

Validation

Similarly to the quantitative model verification, the model validation results have been averaged over $n = 20$ simulation runs in order to account for the stochastic nature of the model. For the PT validation, performance validation has been performed, in which KPI scores have been derived from the historical cool chain data set and compared to the KPI scores obtained from the simulation model. Ideally, performance validation is performed by comparing the model output KPIs against the values obtained from real-world data. However, since the developed OCCE metric is a novel performance measure, it was not possible to derive the three rates from the historic data set. Nonetheless, the handling and cool chain deadlines and the TOR metric have been extracted from the cool chain data set as well as collected from the PT.

5.2.1. First iteration

With regard to the qualitative verification, the model reflected the expected behaviour in line with the presented case study. Therefore, in terms of qualitative verification, the model has passed. The initial verification tests, hypotheses and results have been shown in Table 5.2. As shown, the first four verification tests have failed, thus indicating the necessity for improving the model. In specific, for test one, the R_{OTP} was not below the base scenario value. For tests two and three, the R_{OTP} also failed the test since it was not exceeding the base scenario value, which was expected. Finally, for test four, the R_A was not below the respective base scenario value. In line with the experimental plan in Figure 5.1, not all tests have been passed which led to an investigation into the cause; programming error, input data or the test hypotheses.

Test	Description	Hypothesis	Result	Conclusion
1	$C_{COL} = 12$, $C_{CRT} = 56$	$R_A > 63.04\%$ $R_{OTP} < 94.34\%$ $R_{TA} > 88.50\%$	$R_A = 76.33\%$ $R_{OTP} = 94.60\%$ $R_{TA} = 93.06\%$	Fail
2	$C_{COL} = 3$, $C_{CRT} = 14$	$R_A < 63.04\%$ $R_{OTP} > 94.34\%$ $R_{TA} < 88.50\%$	$R_A = 51.50\%$ $R_{OTP} = 94.23\%$ $R_{TA} = 80.54\%$	Fail
3	$R_{TT} = 16$ hours	$R_A > 63.04\%$ $R_{OTP} > 94.34\%$ $R_{TA} < 88.50\%$	$R_A = 63.76\%$ $R_{OTP} = 94.23\%$ $R_{TA} = 80.45\%$	Fail
4	$R_{TT} = 4$ hours	$R_A < 63.04\%$ $R_{OTP} < 94.34\%$ $R_{TA} < 88.50\%$	$R_A = 63.13\%$ $R_{OTP} = 93.84\%$ $R_{TA} = 88.43\%$	Fail
5	$R_{PCHS} = 2.5$ hours, $R_{KC01} = 1.5$ hours	$R_A < 63.04\%$ $R_{OTP} < 94.34\%$ $R_{TA} > 88.50\%$	$R_A = 61.74\%$ $R_{OTP} = 69.62\%$ $R_{TA} = 89.49\%$	Pass
6	$R_{PCHS} = 10$ hours, $R_{KC01} = 6$ hours	$R_A > 63.04\%$ $R_{OTP} > 94.34\%$ $R_{TA} < 88.50\%$	$R_A = 65.69\%$ $R_{OTP} = 97.95\%$ $R_{TA} = 74.78\%$	Pass

Table 5.2: Initial averaged verification results over $n = 20$ runs

Upon re-evaluation of the PT simulation model, it had become evident that an improvement was necessary in the handling logic of T-ULDs in the case that KC01 storage was full. Therefore, at this point, the cause has been attributed to the model programming. In principle, although the static business ruling for cool storage is in fact in use in the physical system, cool chain operators may have the option to shift around ULDs in the case that KC01 storage capacity utilisation is at a maximum. Until this point in time, this had not been taken into account for the modelling of the PT. Therefore, in an attempt to improve the model behaviour, this dynamic has been incorporated into the current decision-making by assuming that ULDs may be removed in order to make space on the merit of which ULD has the earliest STD. In other words, if an incoming T-ULD should be placed in cool storage according to the business ruling and the storage is full, then the T-ULD with the earliest departure time is removed in order to make space. The latter logic has been visualised in an updated flow chart in Figure 5.6

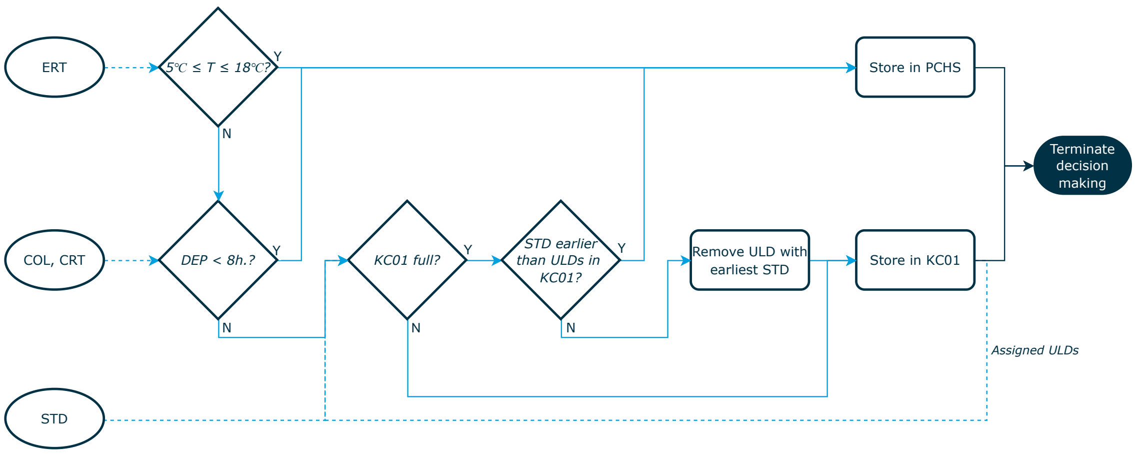


Figure 5.6: Improved cool storage decision-making flow chart based on static business ruling

In order to provide an insight into the model performance at this moment, the initial validation results have been shown in Table 5.3. It can be seen that apart from cool chain deadlines two and three, the percentile deviation from the KPIs obtained from the cool chain data set is limited. In essence, the initial validation results have provided an indication that the Physical Twin appears to be pessimistic with regards to how quickly T-ULDs enter KC01 after arrival (cool chain deadline 2), and optimistic with regards to whether T-ULDs are removed from KC01 no longer than three hours before STD (cool chain deadline 3).

KPI	Data [%]	Physical Twin [%]	Difference
Cool chain deadline 1	96.86	95.03	−1.89%
Cool chain deadline 2	61.86	41.25	−33.32%
Cool chain deadline 3	17.23	25.90	+50.32%
Handling deadline 1	92.74	94.22	+1.60%
Handling deadline 2	92.28	93.25	+0.55%
Handling deadline 3	84.02	89.80	+6.88%
TOR	52.02	52.27	+0.48%

Table 5.3: Initial validation results over $n = 20$ runs

5.2.2. Second iteration

In accordance with the experimental verification and validation plan, the base scenario has been run again given a model programming alteration. This has been reflected in the hypothesis values in Table 5.4 with respect to Table 5.2. The verification tests have likewise been performed again, which has been summarised in Table 5.4. Although an improvement has been made with regard to test one, three verification test failures have remained. Despite the expectation of an increase of R_{OTP} with test two, the on time performance has actually remained equal. Similarly, in test three the R_{OTP} was actually less instead of more than the base scenario rate. Finally, in test four, the R_A was in fact more than what was expected compared to the availability in the base scenario.

Test	Description	Hypothesis	Result	Conclusion
1	$C_{COL} = 12$, $C_{CRT} = 56$	$R_A > 61.13\%$ $R_{OTP} < 95.89\%$ $R_{TA} > 86.06\%$	$R_A = 76.38\%$ $R_{OTP} = 95.46\%$ $R_{TA} = 90.98\%$	Pass
2	$C_{COL} = 3$, $C_{CRT} = 14$	$R_A < 61.13\%$ $R_{OTP} > 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 48.14\%$ $R_{OTP} = 95.89\%$ $R_{TA} = 77.48\%$	Fail
3	$R_{TT} = 16$ hours	$R_A > 61.13\%$ $R_{OTP} > 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 61.74\%$ $R_{OTP} = 95.21\%$ $R_{TA} = 78.12\%$	Fail
4	$R_{TT} = 4$ hours	$R_A < 61.13\%$ $R_{OTP} < 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 61.67\%$ $R_{OTP} = 95.57\%$ $R_{TA} = 85.98\%$	Fail
5	$R_{PCHS} = 2.5$ hours, $R_{KC01} = 1.5$ hours	$R_A < 61.13\%$ $R_{OTP} < 95.89\%$ $R_{TA} > 86.06\%$	$R_A = 59.32\%$ $R_{OTP} = 75.87\%$ $R_{TA} = 88.53\%$	Pass
6	$R_{PCHS} = 10$ hours, $R_{KC01} = 6$ hours	$R_A > 61.13\%$ $R_{OTP} > 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 65.05\%$ $R_{OTP} = 98.29\%$ $R_{TA} = 66.66\%$	Pass

Table 5.4: Second iteration averaged verification results over $n = 20$ runs

In principle, at this point it had been assumed that no programming faults remained in the DES model. Therefore, either the input data or the hypotheses required a revision. In order to test whether the cause was due to the input data, several processing time distributions were altered in order to verify whether this had a significant effect on the model output. However, it has been found that this effect was limited and therefore it has been concluded at this point that the hypotheses required alteration.

The validation results have been collected again with the second verification round and have been shown in Table 5.5. In comparison with the initial validation results, it has been concluded that in principle the degree of model validation has slightly improved with regards to cool chain deadline 3 where the percentile difference between the data and model output has decreased from 50.32% to 47.41%. Nonetheless, a percentile difference of 47.41 points has still been deemed significant. For cool chain deadline 2 no significant improvement has been made. It has been concluded that the significant difference between these two deadlines originates from the fact that the performance could be significantly affected by fresh T-ULDs which are also stored and retrieved from KC01. Besides, it has been concluded that poor data quality could have affected the validation results. Therefore, the respective optimism and pessimism of the model with regard to these cool chain deadlines have remained for future research while the model has been concluded to be partly validated. At the same time, the fact that the model has been partly validated has been recognised as a research limitation.

KPI	**Data** [%]	**Physical Twin** [%]	**Difference**
Cool chain deadline 1	96.86	95.13	−1.79%
Cool chain deadline 2	61.86	41.60	−32.75%
Cool chain deadline 3	17.23	25.40	+47.41%
Handling deadline 1	92.74	93.99	+1.35%
Handling deadline 2	92.28	93.25	+1.05%
Handling deadline 3	84.02	89.67	+6.72%
TOR	52.02	51.74	−0.54%

Table 5.5: Second iteration validation results over $n = 20$ runs

5.2.3. Third iteration

The result of the second iteration had been concluded as an alteration of the hypotheses. Therefore, the base scenario has not been re-evaluated since the model data and programming has not been altered. In specific, after consultation with relevant stakeholders, the following hypothesis changes have been made:

- R_A: For test 4, it has been assumed that the availability remains approximately equal with a halving of the transit time rule. This is due to the fact that, as known from the transit time histogram presented in Section 4.1.3, there is a limited amount of T-ULDs with a transit time shorter or equal to four hours. Therefore, the additional number of ULDs which would require cool storage is limited. For clarity, however, for test 3 the latter is not the case. With a doubling of the transit time rule, it is known from the transit time histogram that there is in fact a significant number of ULDs that no longer receive cool storage, thus leading to an increase of the average availability.
- R_{OTP}: For tests 1, 2, 3 and 4 it has now been assumed that the on time performance remains approximately equal to the base scenario, with the exception of tests 5 and 6. The alteration with regards to on time performance is more in line with reality since the cool storage capacities and transit time rule should in principle not significantly influence whether a ULD makes the connecting flight or not.
- R_{TA}: For test 4, it has been assumed that the temperature adherence remains approximately equal. This is due to the fact that stored T-ULDs with a short transit time are swapped with T-ULDs which have a longer transit time. Furthermore, despite the transit time rule change in tests 3 and 4, the temperature adherence is measured against an allowable exposure of eight hours. Therefore, in contrary to test 3, the effect of halving the transit time rule to four hours on temperature adherence has been expected to be limited.

It should be added that when referring to a test result being approximately equal to the base scenario hypothesis value, it has been assumed that the test result should be within ±1% of the hypothesis value. This is due to the fact that it has been observed in Section 5.1 that small variations in model output may remain with $n = 20$ simulation runs. The resulting verification tests and results for the third iteration have been summarised in Table 5.6. Contrary to expectation, a single verification test failure has remained; test 3. With the assumed limit of ±1% for a test value to be approximate to the hypothesis, it had to be concluded that the result of test 3 was in fact at + 1% and thus not significantly larger as expected. Consequently, test 3 has failed.

Test	Description	Hypothesis	Result	Conclusion
1	$C_{COL} = 12$, $C_{CRT} = 56$	$R_A > 61.13\%$ $R_{OTP} \approx 95.89\%$ $R_{TA} > 86.06\%$	$R_A = 76.38\%$ $R_{OTP} = 95.46\%$ $R_{TA} = 90.98\%$	Pass
2	$C_{COL} = 3$, $C_{CRT} = 14$	$R_A < 61.13\%$ $R_{OTP} \approx 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 48.14\%$ $R_{OTP} = 95.89\%$ $R_{TA} = 77.48\%$	Pass
3	$R_{TT} = 16$ hours	$R_A > 61.13\%$ $R_{OTP} \approx 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 61.74\%$ $R_{OTP} = 95.21\%$ $R_{TA} = 78.12\%$	Fail
4	$R_{TT} = 4$ hours	$R_A \approx 61.13\%$ $R_{OTP} \approx 95.89\%$ $R_{TA} \approx 86.06\%$	$R_A = 61.67\%$ $R_{OTP} = 95.57\%$ $R_{TA} = 85.98\%$	Pass
5	$R_{PCHS} = 2.5$ hours, $R_{KC01} = 1.5$ hours	$R_A < 61.13\%$ $R_{OTP} < 95.89\%$ $R_{TA} > 86.06\%$	$R_A = 59.32\%$ $R_{OTP} = 75.87\%$ $R_{TA} = 88.53\%$	Pass
6	$R_{PCHS} = 10$ hours, $R_{KC01} = 6$ hours	$R_A > 61.13\%$ $R_{OTP} > 95.89\%$ $R_{TA} < 86.06\%$	$R_A = 65.05\%$ $R_{OTP} = 98.29\%$ $R_{TA} = 66.66\%$	Pass

Table 5.6: Third iteration averaged verification results over $n = 20$ runs

Upon further investigation, it has been concluded that the issue should have arisen from the calculation of the availability rate. Therefore, a fourth and final iteration has been performed. Furthermore, since for the third iteration only the hypotheses have been altered, the validation results have not been presented again since it would be a repetition of Table 5.5.

5.2.4. Fourth iteration

For the fourth and final iteration, a programming alteration has been performed. As mentioned in the previous iteration, it had been expected that the issue arose in the calculation of R_A. Although no programming error has been found, it has been found that not enough data was generated during a simulation run for the availability metric. Since R_A is an average quantity, the amount of data points, or individual availability measurements taken during a simulation run, had a limiting effect on the response of the availability rate on an increase of T-ULDs stored in KC01 during test 3. Therefore, apart from extracting an availability data point when a T-ULDs is going in or out of KC01, the availability is now also captured whenever the decision module is activated. Therefore, the extent to which R_A captures the average availability has now been improved. It has been noted that with this programming alteration, only a small addition has been made with regards to KPI data capture, instead of altering any model logic. Nonetheless, the base scenario has been run again in order to reflect the new model alteration, especially with regard to the availability rate. The resulting verification tests and results have been shown in Table 5.7.

Test	Description	Hypothesis	Result	Conclusion
1	$C_{COL} = 12$, $C_{CRT} = 56$	$R_A > 64.68\%$ $R_{OTP} \approx 95.68\%$ $R_{TA} > 85.95\%$	$R_A = 79.63\%$ $R_{OTP} = 95.50\%$ $R_{TA} = 90.87\%$	Pass
2	$C_{COL} = 3$, $C_{CRT} = 14$	$R_A < 64.68\%$ $R_{OTP} \approx 95.68\%$ $R_{TA} < 85.95\%$	$R_A = 48.83\%$ $R_{OTP} = 95.83\%$ $R_{TA} = 77.58\%$	Pass
3	$R_{TT} = 16$ hours	$R_A > 64.68\%$ $R_{OTP} \approx 95.68\%$ $R_{TA} < 85.95\%$	$R_A = 66.75\%$ $R_{OTP} = 95.23\%$ $R_{TA} = 78.35\%$	Pass
4	$R_{TT} = 4$ hours	$R_A \approx 64.68\%$ $R_{OTP} \approx 95.68\%$ $R_{TA} \approx 85.95\%$	$R_A = 64.61\%$ $R_{OTP} = 95.30\%$ $R_{TA} = 85.70\%$	Pass
5	$R_{PCHS} = 2.5$ hours, $R_{KC01} = 1.5$ hours	$R_A < 64.68\%$ $R_{OTP} < 95.68\%$ $R_{TA} > 85.95\%$	$R_A = 62.76\%$ $R_{OTP} = 75.53\%$ $R_{TA} = 88.59\%$	Pass
6	$R_{PCHS} = 10$ hours, $R_{KC01} = 6$ hours	$R_A > 64.68\%$ $R_{OTP} > 95.68\%$ $R_{TA} < 85.95\%$	$R_A = 69.12\%$ $R_{OTP} = 98.35\%$ $R_{TA} = 66.39\%$	Pass

Table 5.7: Fourth iteration averaged verification results over $n = 20$ runs

With regards to test 3, which failed the quantitative verification in iteration three, a pass has now been obtained. With the addition of increased availability data capture, the rate is now able to reflect the expected changes when doubling the transit time rule. Therefore, in terms of quantitative verification, the PT model has now been deemed verified. For the third and fourth iterations, no programming alterations have been made which have had an influence on the model validation. Therefore, the PT model has remained partly validated as indicated in the second iteration in Table 5.5.

5.3. Sensitivity analysis

As indicated in the experimental plan, the final step after verification and validation is a sensitivity analysis. This analysis has been performed in order to study the robustness of the developed DES model. In other words, a sensitivity analysis has been performed in order to investigate the sensitivity of the model with respect to changes in both input data as well as model parameters.

5.3.1. Model parameters

In order to study the model sensitivity to changes in the simulation parameters C_{COL}, C_{CRT}, R_{TT}, R_{PCHS} and R_{KC01}, each parameter has been in- and decreased from its base value. The multiplication has been done through a set of multiplication factors; $F = [0.25, 0.50, 0.75, 1.00, 1.25, 1.50, 1.75]$. In essence, the PT simulation model has been run seven times while averaging the results over $n = 20$ runs, while individually multiplying the respective parameters. Therefore, the effect of changing a single parameter on the model output has been studied. It has been noted that for the cool storage capacity, C_{COL} and C_{CRT} have both been multiplied by the set F at the same time in order to study the model sensitivity to an overall change in cool storage capacity. The latter has also been done for studying the overall sensitivity to the standard storage removal times R_{PCHS} and R_{KC01}. Furthermore, since the COL and CRT cool storage capacities are integers, C_{COL} and C_{CRT} have both been rounded up after each multiplication. The resulting output of the sensitivity analysis has been summarised graphically, indicating the change in model output as the parameters have been subjected to an incremental multiplication factor. Furthermore, the correlation coefficients between the individual rates and the OCCE metric have been determined. The results of the sensitivity analysis have been further discussed below.

The result for the sensitivity analysis of the cool storage capacities C_{COL} and C_{CRT} has been shown in Figure 5.7. The response of each individual OCCE rate as well as the metric itself has been plotted against the multiplication factor increase described by F. In line with the altered verification test hypotheses, it has been found that R_{OTP} is not sensitive to a change in cool storage capacities, indicated by the nearly horizontal profile in Figure 5.7. For the temperature adherence rate R_{TA}, the sensitivity has been found to be most prominent when lowering the storage capacities. This is due to the fact that with reduced capacities, the temperature adherence should logically drop since less ULDs are able to receive the required cool storage in order to meet an allowable exposure time. As the capacities have been increased, however, the slope of R_{TA} has flattened. The flattening occurs since T-ULDs may still exceed the allowable exposure time when not stored in KC01 according to the static business ruling. As an example, excessive ramp time in ambient conditions may result in an exposure excursion of the ULDs. Finally, the sensitivity of R_A approaches a linear profile as expected from the incremental increase of C_{COL} and C_{CRT}. Given the formulation of the availability rate, with an increase in cool storage capacity, the availability should likewise increase as well. However, the profile does show a decreasing slope with an increasing cool storage capacity. This has been attributed to the fact that although capacity has increased, the number of ULDs which have been stored has not increased linearly, since the input data has not been altered. Therefore, the R_A curve has been observed to flatten towards higher cool storage capacities.

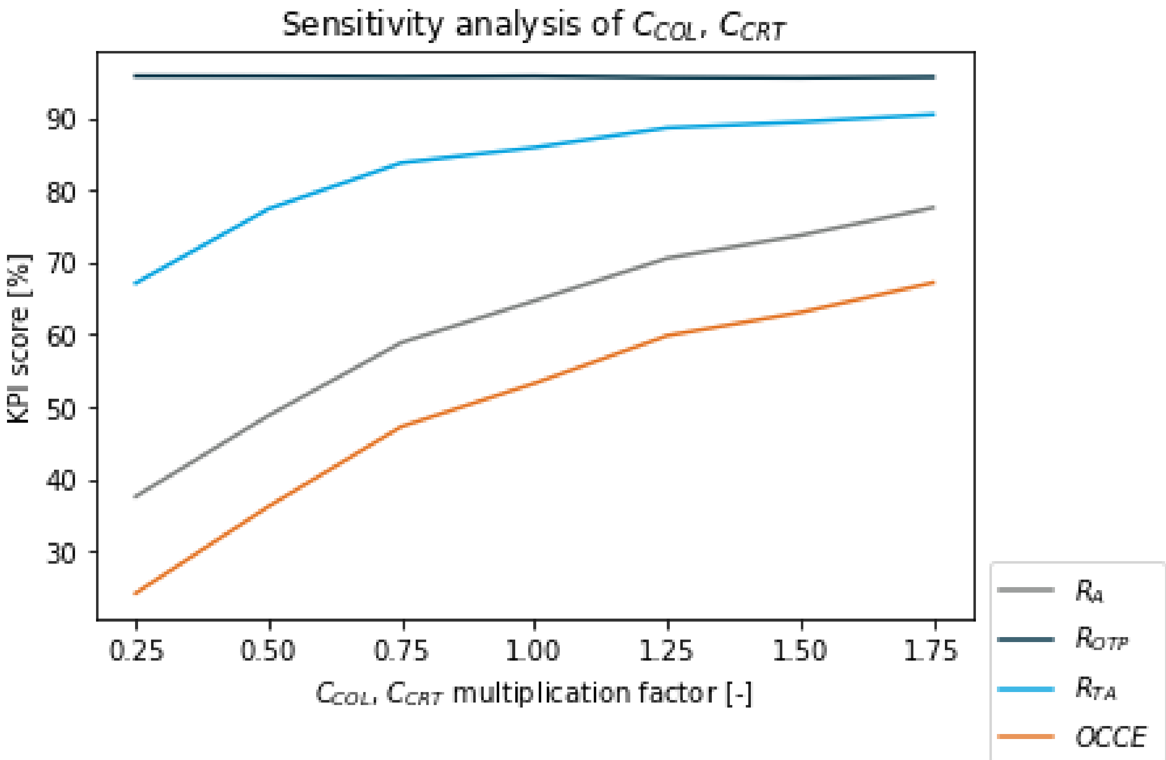


Figure 5.7: Sensitivity analysis of C_{COL}, C_{CRT} using $n = 20$ runs

The result for the sensitivity analysis of the standard storage removal times R_{PCHS} and R_{KC01} has been shown in Figure 5.8. It has become evident that as expected from the quantitative verification, the R_{OTP} is fairly sensitive. This is in line with expectations since a respectively late removal from either storage facility should increase the risk of ULDs missing the connecting flight given the consecutive processing steps before departure. Furthermore, it has been observed that R_{OTP} approaches 100% as the multiplication factor increases, yet does not fully reach it. The apparent asymptotic nature has been attributed to the fact that although late removal from storage should indeed increase the on time performance, it has still been expected that a number of ULDs miss the connecting flight due to other reasons besides the storage removal time. With regards to R_A, a slight linear sensitivity has been observed. As the standard removal times are effectively made earlier as compared to the STD of a T-ULD by means of the multiplication factor set F, the availability is expected to increase since the ULDs spend a shorter time in the cool storage. Considering the R_{TA}, it is evident that the temperature adherence is reduced when ULDs are removed from storage increasingly longer before STD. The individual effect of the different rates has also been visualised, where the R_{OTP} appears to be quite strongly correlated to the OCCE metric, especially for the lower multiplication factors. For higher multiplication factors it appears that the R_{TA} has a more profound influence on the OCCE rate.

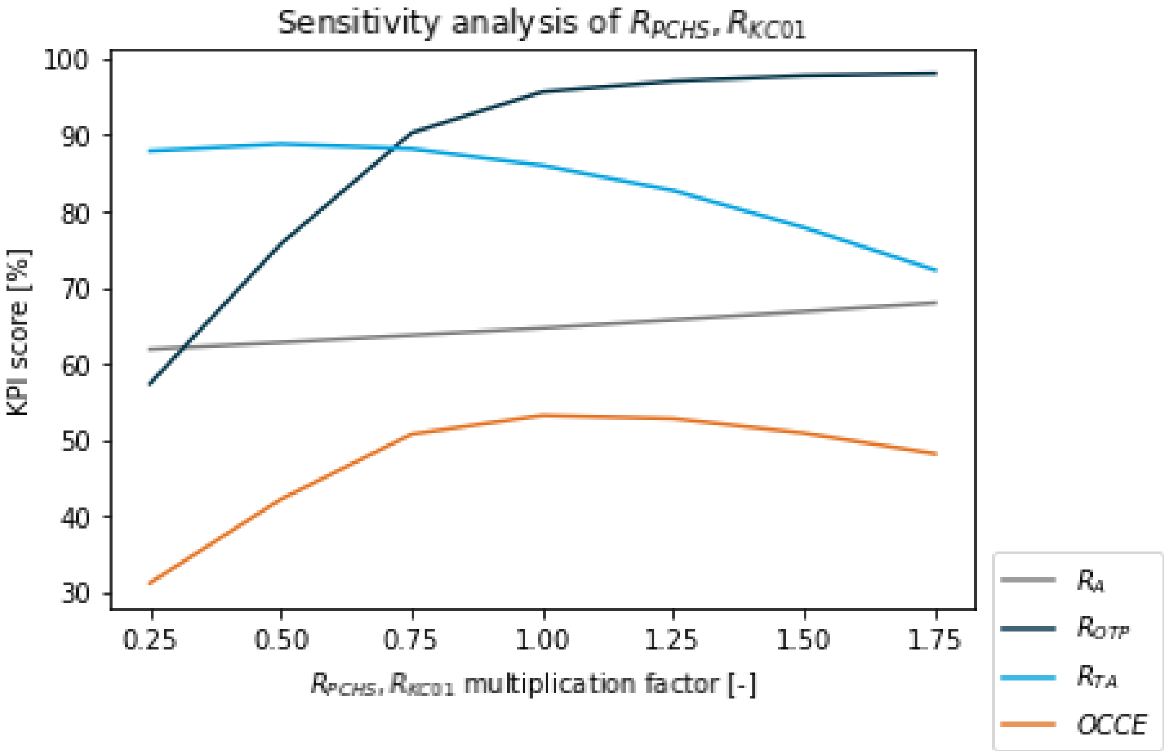


Figure 5.8: Sensitivity analysis of R_{PCHS}, R_{KC01} using $n = 20$ runs

The result for the sensitivity analysis of the transit time business ruling R_{TT} has been shown in Figure 5.9. On a general note, it has been observed that the model output sensitivity to an incremental increase of R_{TT} is limited. In line with what has been found during the verification and validation iterations, R_{OTP} has been found to not be sensitive to changes in R_{TT}, which is in line with expectations from the actual physical system. R_{TA} also appears rather insensitive, although it does show a slight linear increase with an increase of the multiplication factor of R_{TT} of over 1.00. The increase has been expected since with a reduction in the amount of T-ULDs stored in KC01 by an increase of R_{TT}, the availability should also increase. Finally, for R_{TA} the most significant sensitivity has been observed. From a multiplication factor of 1.00 and onwards, a significant reduction in temperature adherence is seen. This reduction is as expected since T-ULDs with a transit time of longer than the eight hours of maximum allowable exposure, do not receive cool storage as R_{TT} has been increased. On the contrary, with a reduction of RTT, the temperature adherence does not increase for the two reasons also covered during the model verification; temperature adherence is still measured against an allowable eight hours and the number of T-ULDs with a transit time shorter than eight hours is limited. Given the fact that R_{TA} exhibits the strongest sensitivity in this analysis, it can also be seen that the OCCE is most strongly affected by temperature adherence.

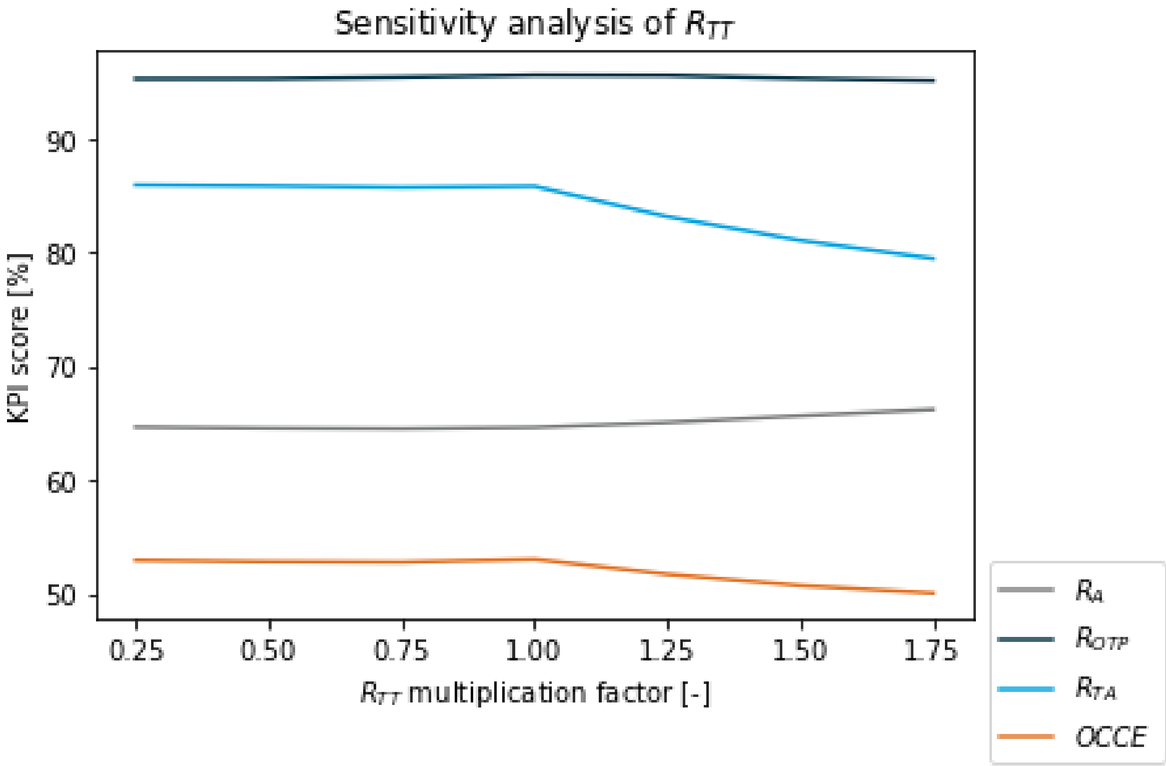


Figure 5.9: Sensitivity analysis of R_{TT} using $n = 20$ runs

In order to study the influence of the respective rates on the OCCE metric in each sensitivity analysis, the Pearson correlation coefficients have been obtained by means of a built-in function in the Pandas package for Python. With the applied function, the linear correlation coefficient is scored between -1 and 1, where -1 indicates a strong negative correlation, +1 strong positive correlation and 0 no correlation. Furthermore, a negative correlation exists when an increase in the one variable is associated with a decrease in the other variable and vice versa. A positive correlation exists when an increase in one variable is associated with an increase in the other variable, or when a decrease is associated with a respective decrease. It has been noted that only the coefficients have been shown of the rates for which a strong correlation has been found since moderate or weak correlation cannot be explained through the sensitivity analysis visualisations.

In Table 5.8, the obtained coefficients have been summarised.The correlation coefficients which have been found for the sensitivity analysis of C_{COL} and C_{CRT} indicate that R_A and R_{TA} have most strongly influenced the OCCE score with a respective coefficient of 1.00 and 0.98. Therefore, both rates are strongly positively correlated, which can be clearly seen in Figure 5.7 since the curves have a similar shape as the OCCE metric. Similarly, for the analysis of the standard storage removal times R_{PCHS} and R_{KC01}, a strong positive correlation has been found for the on time performance rate, which is also seen in Figure 5.8. Consequently, it has been found that R_{OTP} has had the most significant influence on the OCCE. Finally, for the R_{TT} sensitivity analysis, a strong positive correlation coefficient has been obtained for R_{TA}, while a strong negative correlation has been found for R_A, indicating that the OCCE was most strongly influenced by R_A and R_{TA}.

	C_{COL}, C_{CRT}		R_{PCHS}, R_{KC01}	R_{TT}	
Rates	R_A	R_{TA}	R_{OTP}	R_A	R_{TA}
Correlation coefficient OCCE	1.00	0.98	0.96	-0.98	1.00

Table 5.8: Correlation coefficients between the individual rates and the OCCE metric for each parameter sensitivity analysis

5.3.2. Model input data

Similarly to the model parameters, the temperature input data set has been altered incrementally. In specific, the temperature data set has been incremented uniformly by a specific amount of °C. For example, for +5°C, all temperature data points have been increased by 5 °C. The increments have been ranged from −20°C to +20°C in steps of 5°C, where 0°C represents the model output for the actual temperature input data set. The resulting sensitivity analysis data has been visualised in Figure 5.10. As expected, the sensitivity of R_{TA} has been found to be most significant with regard to changes in the temperature input data. Furthermore, a parabola-shaped curve has been observed which confirms the expectation that both very low as well as very high temperatures should affect temperature adherence. It has also been recognised that although the temperature increments are significant, the model response in terms of R_{TA} is relatively moderate. This is due to the fact that temperature adherence is a measure which is based on exposure time, therefore not accounting for the factual temperature deviation from the specified range. Although less sensitive, the same behaviour has been observed for R_A. The effect is limited however due to the static business ruling for cool storage. Therefore, only ERT T-ULDs have attributed to the slight decreases of the availability rate on either side of the unaltered temperature data set since ERT shipments receive cool storage depending also on the weather alarm. Since the weather alarm is issued below 5°C and at or above 18°C, the weather alarm is more prominently issued for more significant temperature data increments. Nonetheless, this effect only influences ERT shipments and thus limits the sensitivity of R_A. Finally, in line with expectations, R_{OTP} has been found to be insensitive to changes in the input temperature data.

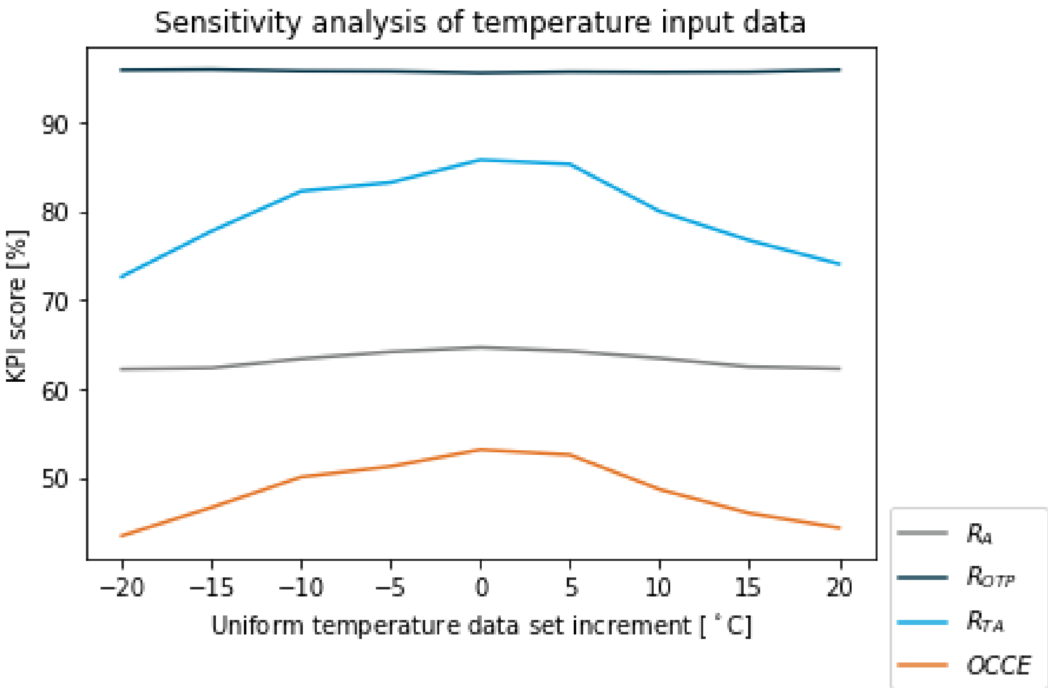


Figure 5.10: Sensitivity analysis of the temperature input data using $n = 20$ runs

Furthermore, the Pearson correlation coefficients for the temperature input data sensitivity analysis have been obtained and summarised in Table 5.9. It has been concluded that R_{TA}, followed by $_{RA}$ has had the most influence on the overall OCCE metric score.

	Temperature input data	
Rates	R_A	R_{TA}
Correlation coefficient OCCE	0.97	1.00

Table 5.9: Correlation coefficients between the individual rates and the OCCE metric for the temperature data sensitivity analysis

5.4. Verification and validation synthesis

In this chapter, an experimental plan has been introduced in order to verify and validate the developed PT model in Chapter 4. As an initial step, the model convergence in terms of the OCCE rate has been studied in order to verify convergence as well as determine an acceptable number of simulation repetitions for model output collection. Consequently, four iterations of qualitative verification, quantitative verification and validation have been performed in order to assure that the model is right and it is the right model for the studied system. Throughout the iterations, several adjustments have been made to the developed model, after which the model has been declared to be verified. Furthermore, sensitivity analyses have been performed on certain key simulation parameters and the temperature input data set in order to verify the sensitivity of the model to changes in input parameters and data. Overall, the sensitivity of the model has been deemed acceptable and in line with expectations. However, in terms of validation, the model has only been considered partly validated. This has been attributed to the poor data quality as well as the set research scope with respect to the collected cool chain data set. Nonetheless, the PT model has been deemed fit for use for the implementation of the proposed DT improvement concept and the collection of the results, which has been addressed in the following chapter.

6

Model implementation

Following the verification and validation of the PT simulation model, in the following chapter, the DT improvement concept has first been discussed in Section 6.1. Consequently, the resulting model output has been presented and discussed in Section 6.2. Finally, an answer has been provided to the following research questions:

5 How can the improved decision-making be implemented?
6 Which insights can be derived from the developed model?

6.1. Decision support module

For the model implementation, the whole digital system has now been considered, consisting of the PT with the addition of the DT. The digital system has been visualised in Figure 6.1. In the process of replicating the actual cool chain behaviour by the PT, data is generated which consequently can be used by the DT part of the digital system. This data includes the current COL and CRT cool storage availability, the cumulative exposure of the individual ULDs, the temperature at different points in the system and the time until departure. As indicated in the conceptual model in Figure 4.11, the proposed cool chain management improvement has been materialised in the form of the DT; a decision support module. As opposed to the static business rules deployed in the cool storage decision-making process, such a decision support module has been envisioned to be able to improve the decision-making based on real-world and digitised data provided by the PT and physical system data extraction. The interaction between the PT model and DT as visualised in Figure 6.1 is comprised of the exchange of system and ULD level data. This information is then utilised in the DT, after which cool storage decisions are determined and fed back to the PT. By enabling the DT and thus the decision support module, changes are inferred in the PT model, for which the resulting output can be gathered while quantifying possible improvements by means of the OCCE metric.

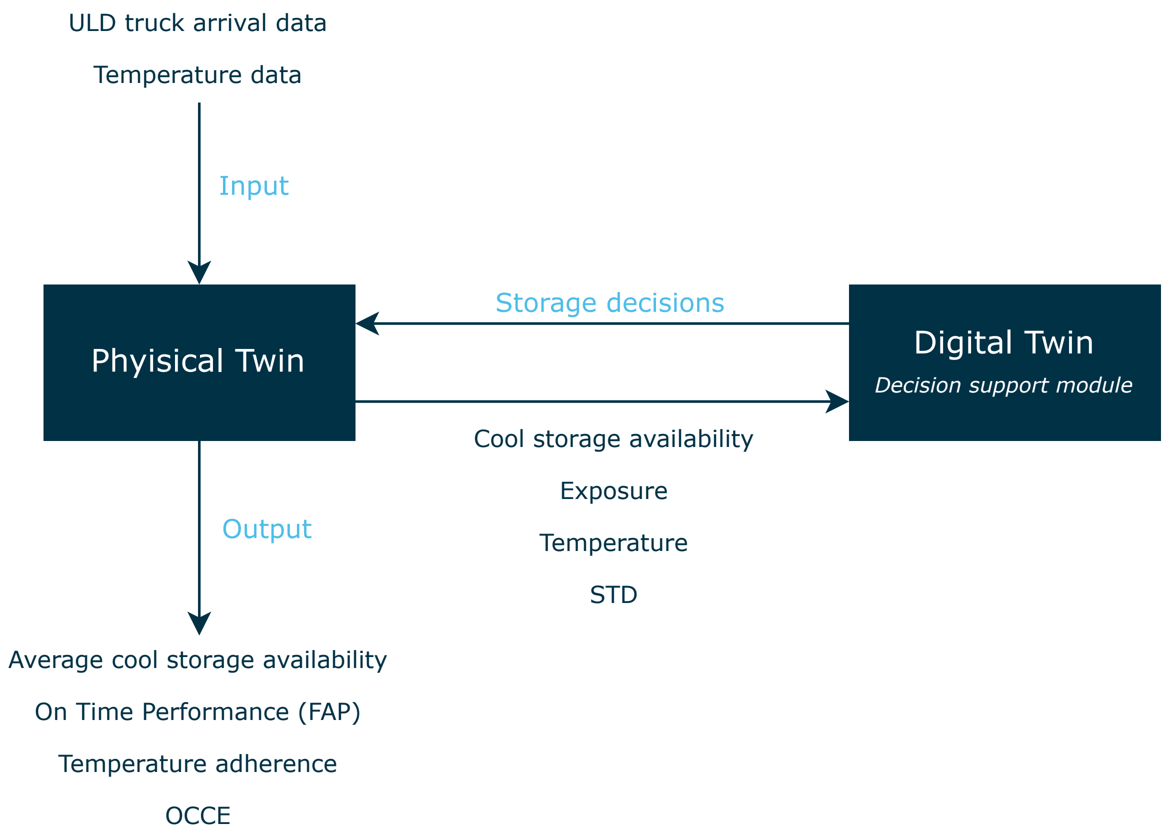
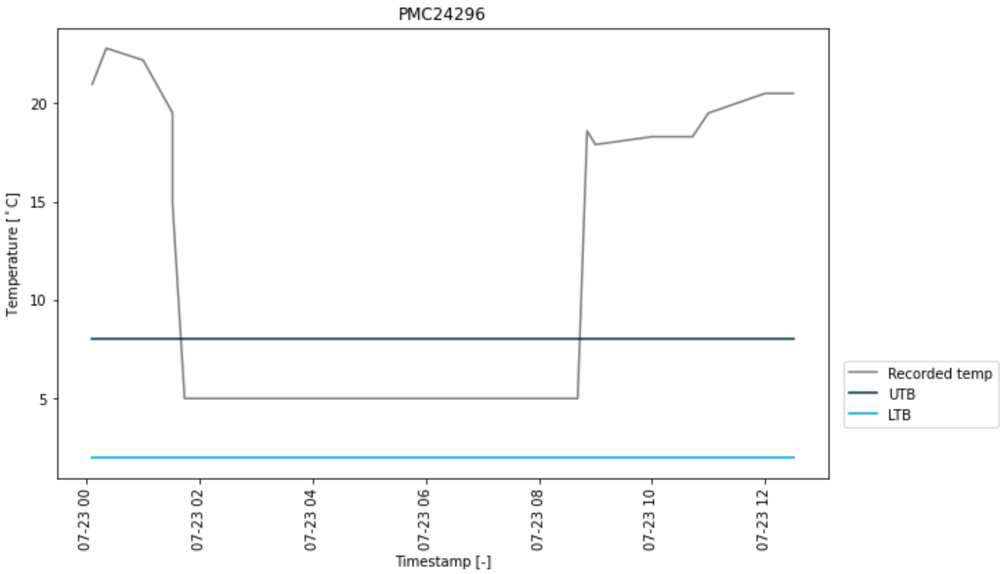
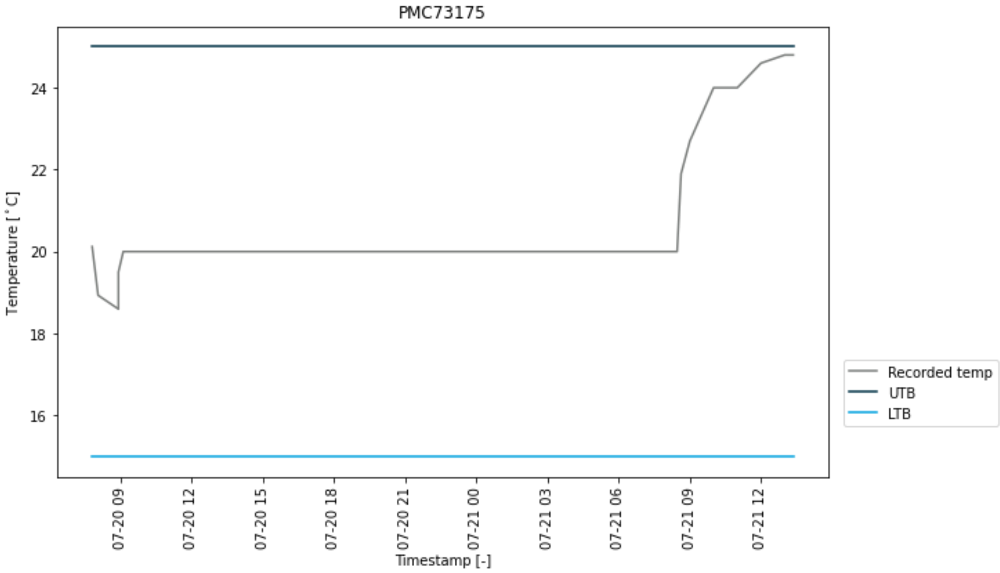


Figure 6.1: Schematic overview of the proposed digital system

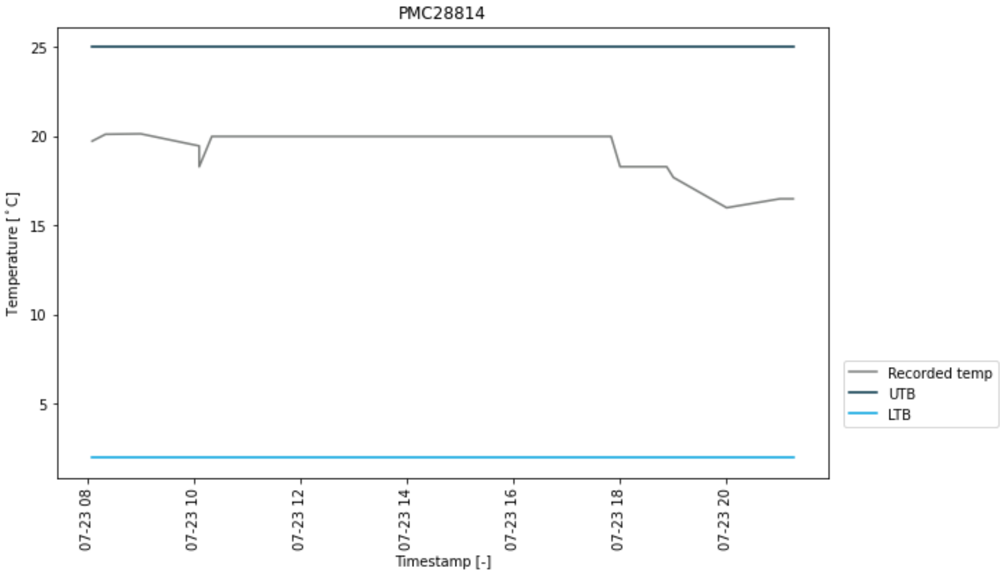
Among other purposes, the input temperature data in the PT has allowed the generation of exposure plots for each T-ULD which went through the system. An example has been given for three ULDs with the three respective SHCs in Figure 6.2. Throughout the process, temperatures have been logged with a corresponding timestamp and stored for each ULD, which enables the visualisation of the temperature profile throughout the handling process, as well as the determination of the cumulative exposure at a specific time. Given the typically long transit times of ULDs, rapidly changing temperature spikes may occur for instance as a ULD is briefly outside while being transported to KC01. Nonetheless, the exposure plots may provide a useful method for the monitoring of temperature-sensitive freight. Furthermore, as seen in Figure 6.2c, this specific ERT T-ULD has been stored in the CRT cool storage room since the ambient temperature was above 18°C. However, from the temperature profile, it could be concluded that storage in the PCHS would have sufficed since the recorded temperature would have likely remained within the designated UTB and LTB. It has also been noted that the straight horizontal segments in the temperature profiles represent the time spent in cool storage at the specified temperature set points. Although realistically variations in temperature would still occur, for this research it has been assumed that the temperature remains constant in the cool storage facility.



(a) Exposure of a COL T-ULD



(b) Exposure of a CRT T-ULD



(c) Exposure of an ERT T-ULD

Figure 6.2: ULD exposure plots extracted from the PT simulation model

Apart from the exposure data, other relevant information is extracted from the PT and fed into the DT. A functional overview of the DT decision support module has been provided in Figure 6.3, where the arrows indicate the functional relations. Similar to the PT, the DT has also been manually programmed in Python, for which the code can be seen in Appendix B.2. It has been implemented in a separate module, which is imported into the main PT program.

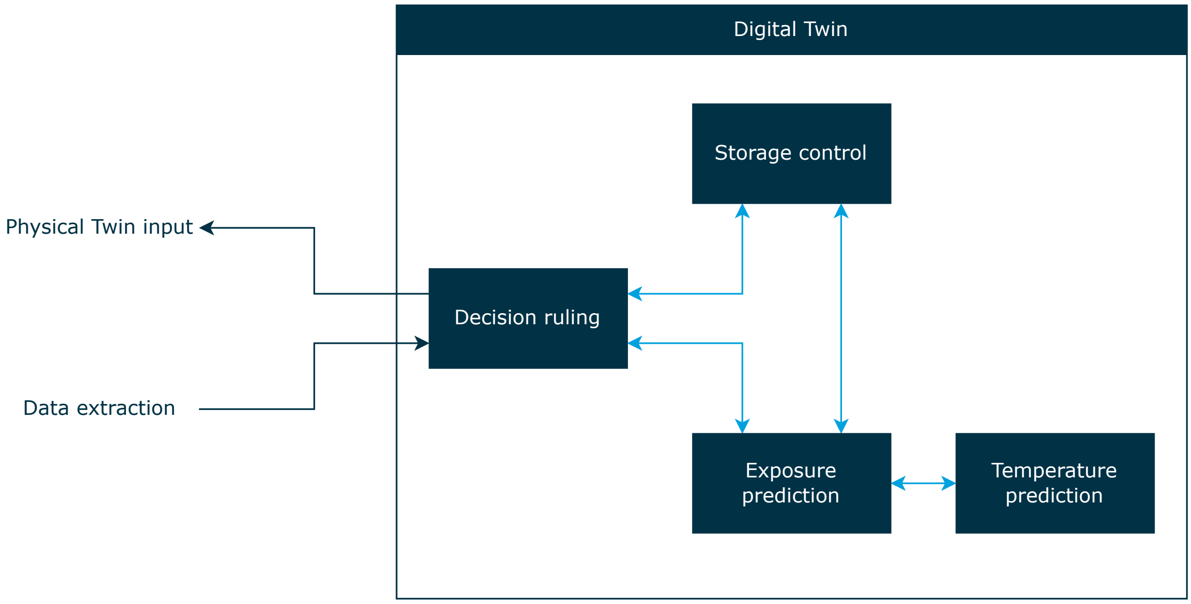


Figure 6.3: Functional components of the proposed DT decision support module

In principle, the DT has been built up from four primary functional components, which have been further described:

- **Decision ruling**: the dynamic ruling for the cool storage decision has been implemented in this component. The decision ruling component ultimately returns the storage decisions which are fed back to the PT. However, before the determination of the decision, the other components have been utilised.
- **Exposure prediction**: the predicted exposure of a T-ULD has been taken as the basis for dynamic decision-making. In essence, this component predicts the expected exposure when storing the ULD in KC01 or the PCHS. Depending on the predicted exposure, the storage decision is consequently made.
- **Temperature prediction**: in order to determine the predicted exposure, the temperature prediction component provides the expected ambient and PCHS temperatures. Therefore, this component can be compared to having accurate temperature forecasts for ambient temperatures and a temperature prediction functionality for the PCHS temperature based on historical data, along with the current temperatures in the system.
- **Storage control**: finally, in the case that the best decision is to store a ULD in KC01 while there is insufficient cool storage availability, the storage control component determines whether to remove a ULD from storage in order to make room for the new ULD. This has been implemented based on the selection of the ULD in cool storage with the earliest STD and the lowest cumulative exposure. Consequently, in order to determine the storage decision, the predicted exposure component is used in order to estimate which choice would lead to the lowest overall exposure for both ULDs. In the case that a ULD has been selected to be removed from cool storage, the decision ruling component feeds back both the decision for the new ULD as well as the ULD which is to be removed.

The described process of interaction between the different components has been further visualised in a flow chart, as seen in Figure 6.4. The data input, as well as exchange between the different functional components in the DT, has been indicated by the striped arrows whereas the solid arrows indicate the decision-making logic flow. Furthermore, the parts in the process where the cool storage and removal decisions are made have been annotated in light blue.

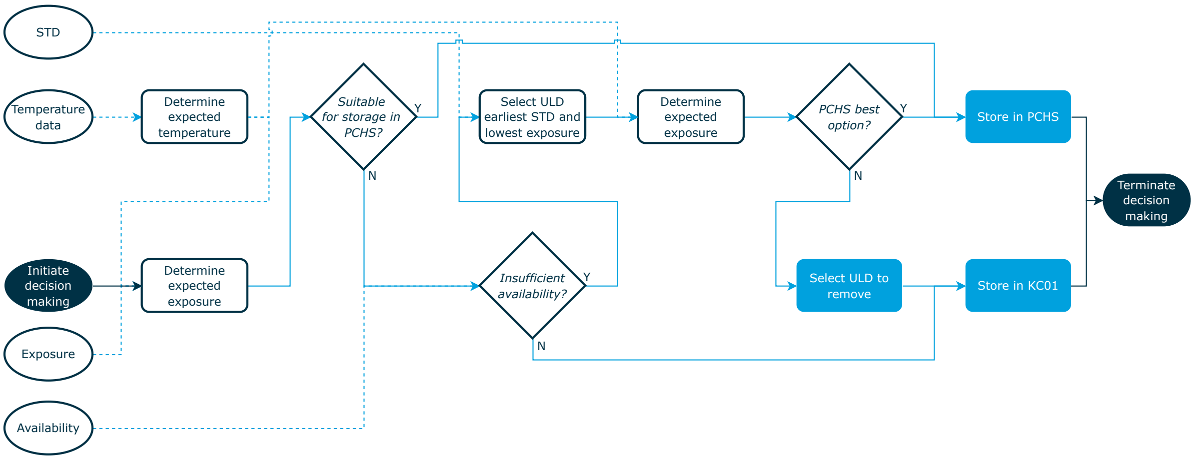


Figure 6.4: DT dynamic decision making flow chart

Ultimately, by enabling the DT module, the static business ruling has been replaced by dynamic decision support which determines the cool storage decisions based on the actual system state as well as the cumulative exposure of the T-ULDs. Since an actual implementation of the proposed digital system into the physical system has not been accomplished in this research, further recommendations for implementation have been provided in Section 7.2. Firstly, however, the results from the developed model have been collected, presented and discussed in the following section.

6.2. Results

As previously described, the PT output includes the proposed OCCE operational quality metric along with the respective rates from which it has been built up. The case study, as presented in Chapter 3 and verified and partly validated in Chapter 5, has been used as the baseline scenario with which the proposed DT concept has been compared. Firstly, the general results for the total simulation duration of one year have been described. Consequently, the model output and performance comparison has also been studied for three seasons of interest throughout the studied year.

6.2.1. General results

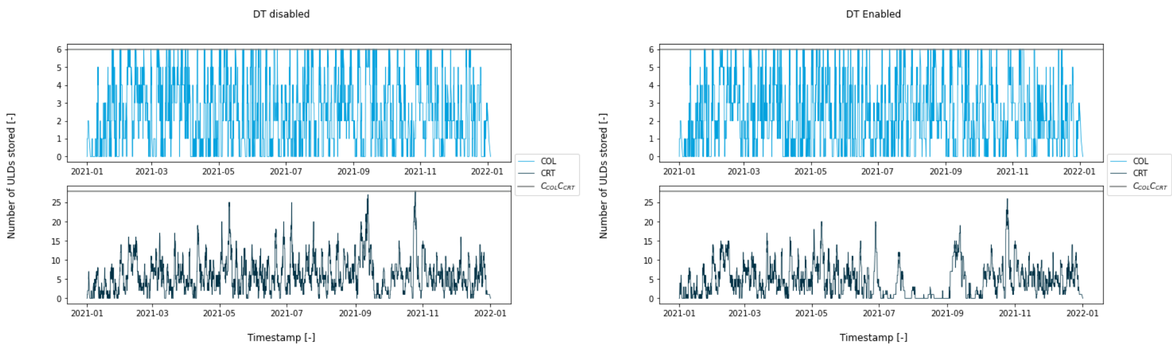
The general results include the digital system output as collected for the whole simulation duration of one year with $n = 20$ repetitions and have been summarised in Table 6.1. For each individual rate as well as the OCCE, the results have been shown for the baseline scenario and the scenario in which the DT has been implemented. Furthermore, the difference in performance between baseline and DT implementation has been shown. Moreover, although not included in the OCCE metric, the average exposure per ULD has also been extracted in order to provide additional insights into the effect of the DT implementation.

KPI	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [hh:mm:ss]
Baseline	54.70	74.66	64.68	95.68	85.95	53.19	04:10:41
DT implementation	55.21	80.34	67.87	95.04	85.59	55.21	04:05:32
Difference	+0.93%	+7.61%	+4.93%	-0.67%	-0.42%	+3.80%	-2.05%

Table 6.1: Physical twin model output results with the DT implementation for the full simulation duration averaged over $n = 20$ runs

From Table 6.1, several observations have been made. In general, the OCCE has increased by 3.80%, which has been achieved through an increase of R_A of 4.93%. Moreover, it has been found that through the DT implementation, primarily the $R_{A,CRT}$ has contributed to the overall cool storage availability increase. This can be attributed to the fact that CRT and ERT T-ULDs are in general more suitable for PCHS storage with regards to the acceptable temperature range. Besides the availability, it has been observed that both R_{OTP} and R_{TA} have slightly decreased by 0.67% and 0.42% respectively. In principle, any decrease in either rate has been deemed as unwanted given the costliness of pharmaceutical air freight. However, given the minor variance observed in the convergence analysis up to $n = 30$ runs for both rates, the decreases have been considered insignificant. Finally, it has been concluded that the average T-ULD exposure has been slightly decreased through the implementation of the DT improvement concept.

Apart from the model output, the number of ULDs stored in KC01 throughout the simulation duration of one year has been shown for a single run in Figure 6.5. By comparing Figure 6.5a with Figure 6.5b, it has been observed that in line with the availability results, the primary influence in terms of cool storage decisions have been made on the CRT cool storage room. The most significant impact has been observed between 2021-07 and 2021-09, where the number of stored T-ULDs has been significantly decreased while maintaining a roughly equal temperature adherence and slightly decreased average exposure. With regards to the COL storage room, no significant changes can be observed from Figure 6.5. This has been attributed to the fact that the COL temperature range between 2°C and 8°C is more stringent while the temperature in the PCHS is generally more suitable for CRT and ERT T-ULDs.

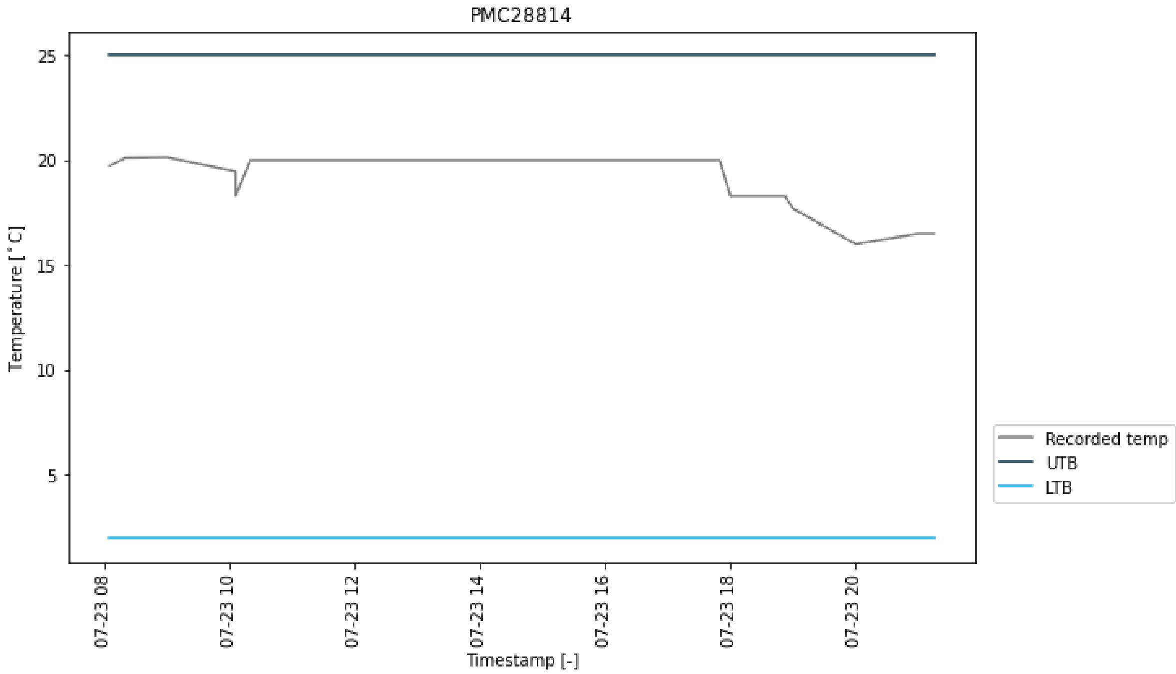


(a) Disabled DT module

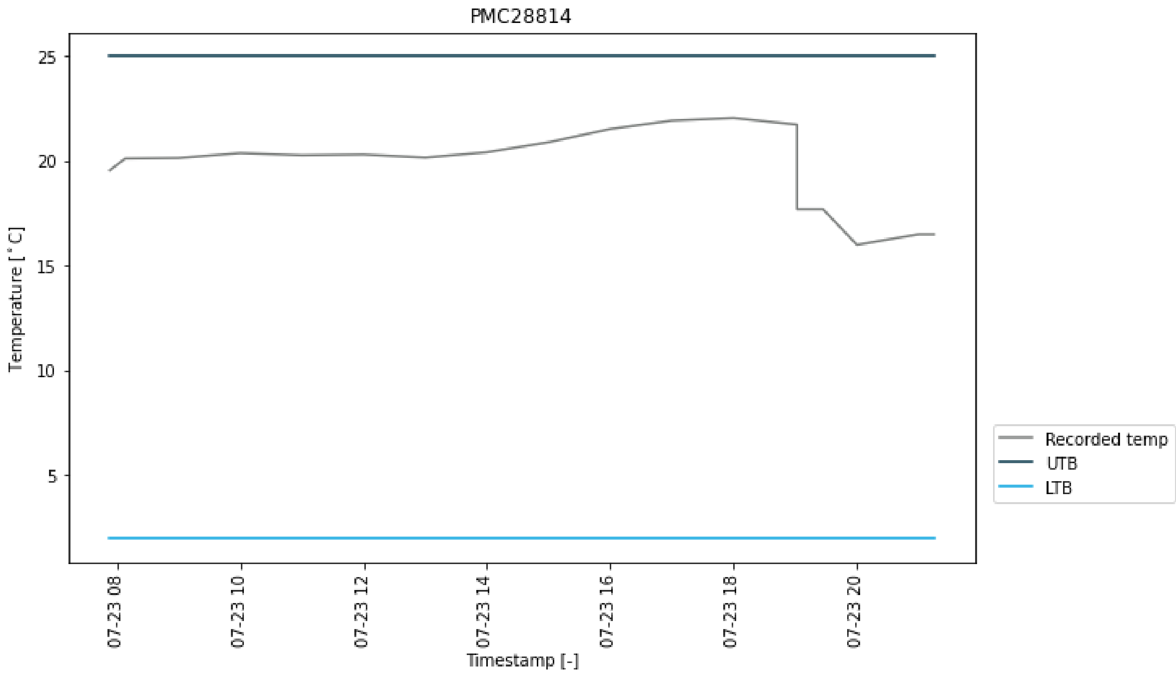
(b) Enabled DT module

Figure 6.5: Number of ULDs stored over the total duration of a single simulation run

In order to highlight the effect of the DT implementation, the exposure plot of an ERT T-ULD has been shown in Figure 6.6 with and without the decision support module enabled. From Figure 6.6b it can be seen that the ULD was indeed suitable for PCHS storage, leading to no exposure throughout the processing time. Therefore, the ULD has been correctly handled well within the respective UTB and LTB without utilising any additional resources such as cool storage as well as additional movements to and from the facility. Additionally, with respect to cool chain monitoring, the digitisation of the studied system has shown that the possibility exists to utilise currently available system data into temperature profile plots for each unique T-ULD. Such temperature plots would enable enhanced insight into the exposure endured by individual ULDs throughout the cool chain and could potentially be used for alerting functionalities with respect to the maximum allowable exposure.



(a) Disabled DT module



(b) Enabled DT module

Figure 6.6: Comparison of an ERT T-ULD temperature profile with and without DT implementation

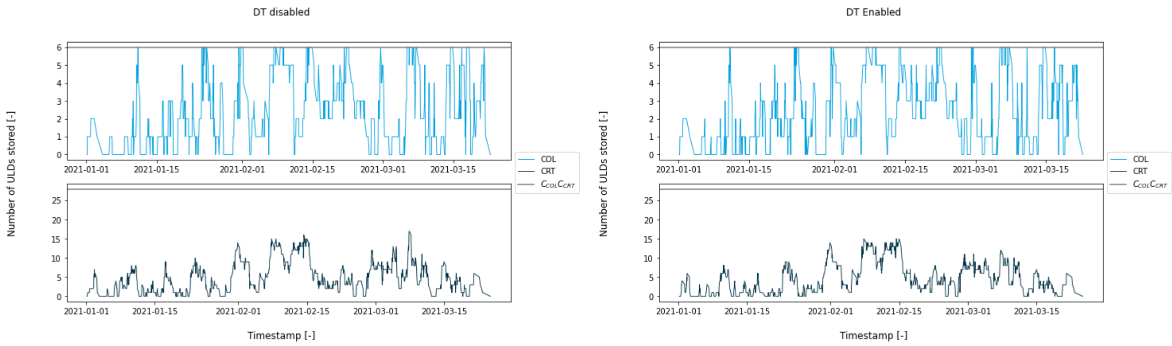
6.2.2. Winter results

In a similar manner, the results have been obtained for the initial winter period of the first three months of the year with $n = 20$ repetitions. The resulting scores have been summarised in Table 6.2.

KPI	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [hh:mm:ss]
Baseline	57.04	80.35	68.54	96.37	87.84	58.02	04:34:40
DT implementation	57.89	81.48	69.29	95.64	87.77	58.16	04:32:35
Difference	+1.50%	+1.41%	+1.09%	-0.76%	-0.08%	+0.24%	-0.76%

Table 6.2: Physical twin model output results with the DT implementation for the winter period averaged over $n = 20$ runs

From Table 6.2, it has been observed that the relative contribution to the overall availability increase is approximately equal for both COL and CRT for the winter period at the beginning of the year. The significant contribution from $R_{A,CRT}$ has likely decreased given the fact that the PCHS temperature in this period is more likely to be unsuitable for CRT storage. Furthermore, R_{OTP} has been observed to slightly decrease for this period while R_{TA} has remained approximately equal. Additionally, the average exposure has slightly decreased following the DT implementation. Finally, it has been concluded that the OCCE metric has only been slightly increased in the winter period. The limited impact on the cool storage has also been observed in Figure 6.7, where there is no clear difference between Figure 6.7a and Figure 6.7b respectively.



(a) Disabled DT module

(b) Enabled DT module

Figure 6.7: Number of ULDs stored during winter of a single simulation run

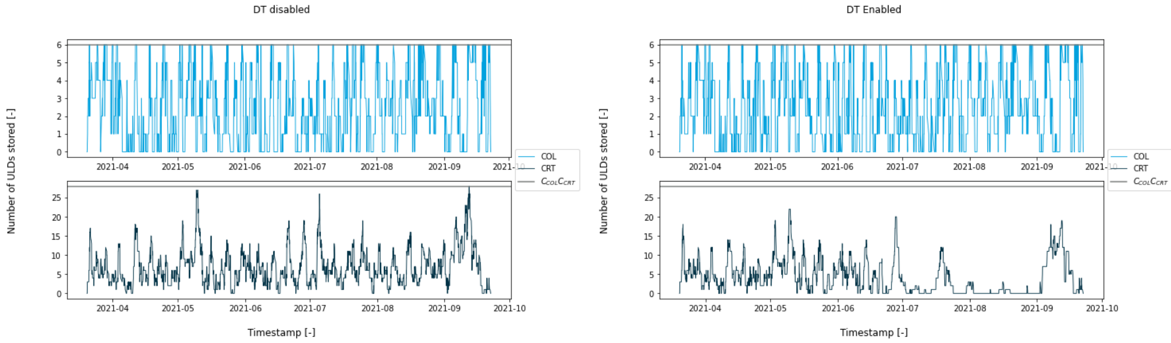
6.2.3. Spring and Summer results

In principle, the spring and most importantly summer periods can be considered as the most critical periods during the year for a pharmaceutical air freight cool chain. For this period, the results which have been gathered over $n = 20$ runs have been summarised in Table 6.3.

KPI	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [hh:mm:ss]
Baseline	51.84	72.38	62.22	95.62	85.76	51.02	03:52:59
DT implementation	52.61	80.88	66.78	94.98	85.58	54.28	03:49:00
Difference	+ 1.49%	+ 11.74%	+ 7.33%	- 0.67%	- 0.21%	+ 6.39%	- 1.71%

Table 6.3: Physical twin model output results with the DT implementation for the spring and summer period averaged over $n = 20$ runs

As expected, a significant increase of R_A has been observed in the spring and summer period, which has been mostly caused by a significant increase in $R_{A,CRT}$ of 11.74%. Therefore, a significant amount of CRT and ERT T-ULDs can safely be stored in the PCHS during a critical period. Furthermore, a slight decrease has been found again for R_{OTP} and R_{TA} while the OCCE has significantly increased due to the cool storage availability increase. Finally, the average exposure has been found to be decreased. In Figure 6.8, the significant effect on $R_{A,CRT}$ can evidently be seen through the comparison of Figure 6.8a and Figure 6.8b. Interestingly, it appears that the most significant impact has been made between 2021-07 and 2021-09.



(a) Disabled DT module **(b)** Enabled DT module

Figure 6.8: Number of ULDs stored during spring and summer of a single simulation run

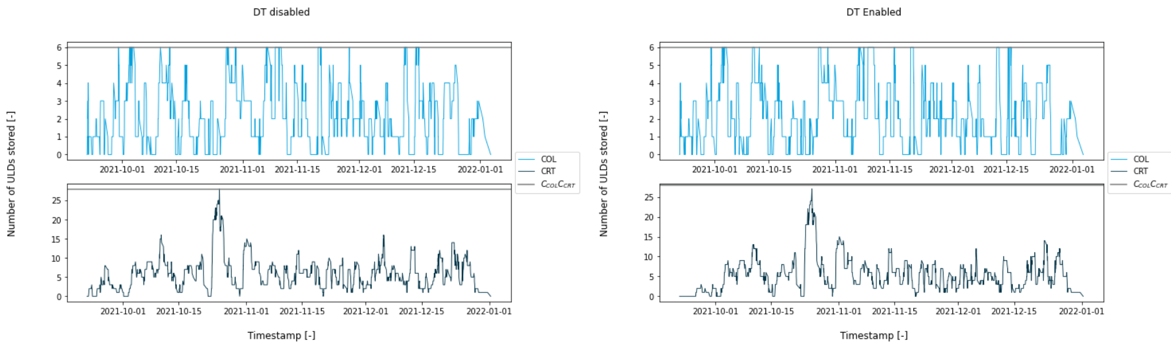
6.2.4. Autumn and Winter results

Finally, the results have been gathered for the subsequent autumn and winter period and have been summarised in Table 6.2.

KPI	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [hh:mm:ss]
Baseline	60.87	75.72	68.45	96.16	87.82	57.80	04:30:46
DT implementation	61.28	78.27	69.90	95.36	87.80	58.53	04:24:25
Difference	+0.67%	+3.37%	+2.12%	-0.83%	-0.02%	+1.26%	-2.35%

Table 6.4: Physical twin model output results with the DT implementation for the fall and winter period averaged over $n = 20$ runs

Compared to the winter period at the beginning of the year, the contribution of $R_{A,CRT}$ to the overall availability increase has been found to be slightly larger at 3.37%. Similarly to the other periods, the R_{OTP} has slightly decreased while R_{TA} has remained approximately equal, resulting in a slight increase of the OCCE. Finally, the average exposure has been decreased by 2.35%. As expected from the gathered data shown in Table 6.4, no significant impact on the number of ULDs stored has been found in the visualisation in Figure 6.9. Nonetheless, comparing Figure 6.9a and Figure 6.9b respectively, a slight decrease of T-ULDs stored in the CRT room can be observed especially a the beginning of autumn, up to the beginning of 2021-10.



(a) Disabled DT module **(b)** Enabled DT module

Figure 6.9: Number of ULDs stored during autumn and winter of a single simulation run

6.2.5. OCCE performance differences

In order to provide an overview of the obtained differences with respect to the baseline scenario, the differences have been summarised for the general case as well as the different accentuated seasons and can be seen in Table 6.5.

	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [%]
General	+0.93	+7.61	+4.93	-0.67	-0.42	+3.80	-2.05
Winter	+1.50	+1.41	+1.09	-0.76	-0.08	+0.24	-0.76
Spring - Summer	+1.49	+11.74	+7.33	-0.67	-0.21	+6.39	-1.71
Autumn - Winter	+0.67	+3.37	+2.21	-0.83	-0.02	+1.26	-2.35

Table 6.5: PT model including the DT module output differences compared to the baseline scenario for the different periods throughout the year

Several findings have been gathered from the resulting overview in Table 6.5. Firstly, in general, it has been found that the operational quality of the studied cool chain can be improved through the utilisation of actual system data such as temperature in order to replace the static business ruling used for cool storage decisions towards more dynamic business ruling based on the expected exposure of each ULD. The improvement has primarily been obtained through the improvement of the cool storage availability, especially for the CRT cool room. In fact, this improvement has been found to be most significant in one of the critical periods of the year for an air freight pharmaceutical cool chain: spring and more importantly summer. In effect, the increased availability could either be utilised for an increased freight volume or be used to possibly store COL freight if the cool storage facility could be arranged flexibly with respect to temperature. With regards to the COL storage room, it has been found that the availability has increased more significantly in the first winter, spring and summer periods. More generally, it has been found that fewer resources in terms of cool chain equipment would be required while maintaining roughly an equal temperature adherence and decreasing the average ULD exposure. Secondly, it has been observed that the R_{OTP} has shown a decrease across all specified periods. In principle, the magnitude of the decrease in the on time performance has been deemed insignificant due to the observation of variance in this rate in the convergence analysis. Nonetheless, it has been noted as a recommendation for further research and a possible point of improvement. Thirdly, similarly to the on time performance, R_{TA} has been found to slightly decrease. However, the magnitude has been found to be limited, especially for the autumn and winter periods. Furthermore, a slight variance has been observed for the temperature adherence in the convergence analysis for $n > 20$. Finally, given the fact that the availability increase for CRT is most significant during spring and summer and for COL during the first winter period, it could indicate the possibility to implement flexible cool storage capacities which could facilitate multiple temperature zones throughout the year. As an example, cool storage capacity for CRT and ERT freight during summer could be reduced in order to provide additional COL cool storage capacity. However, it has been concluded that additional research would be required in order to adequately determine and or control flexible cool storage capacities, as well as a control scheme for suitable PCHS storage locations. Nonetheless, the developed cool chain improvement methodology has provided the required modelling groundwork for such research.

6.2.6. Results synthesis

In this chapter, the actual DT part of the proposed digital system for cool chain operational quality improvement has been introduced and implemented by connecting it to the PT model. Consequently, the results have been gathered by comparing the performance of the baseline case study system to the performance with the implemented DT concept. In the following and final chapter, a final conclusion has been given along with recommendations for further research.

7

Research conclusion

In this final chapter, firstly the research conclusion has been discussed by considering the answers to the sub-questions in Section 7.1 in order to provide an answer to the main research question:

> To what extent can the pharmaceutical cool chain operational quality be improved through the development of a real-time decision-making methodology?

Consequently, in Section 7.2, recommendations for further research have been provided in order to highlight the steps to be taken towards the actual implementation of the proposed cool chain improvement methodology.

7.1. Conclusion

In the following and final section, an answer has been formulated to the research questions which have guided the carrying out of this research project.

1. Considering the state of the art, how can cool chain management be improved?

It has been acknowledged that information extraction and improved decision-making may provide a suitable method for cool chain management, where the DT concept has been recognised as a suitable method. First, however, a discussion has been given on missing data imputation, which is typically required during the investigation of industry data. Then, in order to quantify any cool chain management improvements, the OEE methodology has been adapted into the proposed novel OCCE operational quality metric. Consequently, the different available modelling techniques have been studied in order to select an appropriate method for the presented research project. In line with the findings from the cool chain management literature survey, the DT concept has been introduced, defined and finally presented as the proposed improvement method. Therefore, considering the state of the art, the DT concept has been chosen as an appropriate method for cool chain management improvement.

2 What is the current state of a pharmaceutical air freight cool chain process, based on an applied case study?

It has been found that a multitude of different KPIs are in use. However, the lack of a single metric which encompasses the operational quality of a part of a cool chain or the system as a whole such as the proposed OCCE metric has been acknowledged. Furthermore, the decision to store freight in cool storage is dependent on the static business ruling which may not be optimal considering the current environmental conditions, despite significant constraints on cool storage capacities. Besides, it has been recognised that although there is a significant data capture from for instance the WMS in use, the data has not been utilised in the monitoring of freight conditions such as exposure. Therefore, there are significant steps which can be taken in terms of the digitisation of the cool chain process in the air freight industry. In conclusion, it has thus been noted that pharmaceutical air freight cool chains remain highly suitable for the digitisation of processes for further performance improvements.

3 How can a pharmaceutical air freight cool chain be modelled?

It has been concluded that a pharmaceutical air freight cool chain can be modelled by means of DES. Furthermore, the importance of data collection and handling has been underlined with respect to ultimately the model quality as well as the required time and effort necessary to obtain an adequate model input in terms of data.

4 To what extent does the developed model effectively represent the cool chain process, in terms of verification and validation?

With respect to the verification, the model has been considered to be verified after four iterations of the experimental plan, during which several changes have been made to the modelling as well as the verification hypotheses. With regard to model validation, the model has been considered to be partly validated since an acceptable percentile difference between the data set and model output has not been obtained for two cool chain deadline KPIs. Consequently, the model has been acknowledged to be pessimistic with regard to the timeliness of ULDs entering KC01 and optimistic in terms of the standard removal time from KC01. In conclusion, the developed model thus represents the cool chain process in terms of verification and partly represents the process in terms of validation.

5 How can the decision-making be implemented?
6 Which insights can be derived from the developed model?

In conclusion, the improved decision-making has been implemented through the programming of a separate module which has been imported into the PT model. Furthermore, it has been found that during spring and summer, significant improvements can be made with respect to the CRT storage room through the dynamic business ruling scheme in the DT module. Besides, flexible cool storage capacities have been recognised as a potential improvement and area of interest for future cool chain systems since the availability improvement appears to be season dependent. In other words, the additional availability obtained for the CRT room could potentially be utilised for COL freight. In addition, it has been found to be possible to reduce the average exposure of ULDs in the cool chain system.

To what extent can the pharmaceutical cool chain operational quality be improved through the development of a real-time decision-making methodology?

With regard to the cool chain operational quality as quantified through the OCCE metric, it has been found that in general over the studied period of a year an improvement of 3.80% can be made. However, during the critical spring and summer periods, the operational quality can be improved by 6.39%, largely attributed to the increase of CRT cool storage availability. For the first winter and fall and winter periods, slight OCCE improvements have been found with respectively 0.24% and 1.26%. It has also been found that the proposed improvement method has not been able to provide an improvement with respect to on time performance and temperature adherence, although the respective decreases of 0.67% and 0.42 % could be considered insignificant taking into account the performed convergence analysis. In conclusion, a quantifiable improvement of the pharmaceutical cool chain operational quality has been obtained through the developed methodology.

7.2. Recommendations for further research

With regard to the convergence analysis, it has been observed that for the on time performance and temperature adherence, slight variations remained in the PT model output. Therefore, the relative variance in these rates has been noted as a point of improvement in terms of the DES modelling. Following the convergence analysis, verification and validation have been performed for the physical twin model. However, only partial validation has unfortunately been achieved, which has been acknowledged as a limitation of the performed research. The partial validation has resulted in a respective pessimism with regard to the time it takes for T-ULDs to be stored in KC01 after arrival, and an optimism with regards to the timeliness of removal from cool storage three hours before departure.

Although it has been found that the proposed improvement method by means of the digital system is able to improve the cool chain operational quality, the actual implementation into a cool chain should also be considered. In general, further research has been recommended with regard to the integration and or connection of the proposed digital system with the systems in use in the respective cool chain.

As an example, for the presented case study this entails that data should automatically be drawn from relevant data sources such as the WMS Chain and in the future CargoBus, as well as the ATAL temperature sensor network. However, especially with regards to the poor data quality as well as the missing data which has been encountered throughout the research project, further research has also been recommended for the issue of reliable data collection and sensor redundancy. In general, Coelho, Relvas, and Barbosa-Póvoa [108] have mentioned that companies should invest heavily in the improvement of infrastructure and systems, especially in the installation of sensors for real-time data collection. It has been argued that only then it is possible to represent reliable real-time simulation models. This is due to the fact that the digital system requires accurate and real-time data in order to mirror the functions, behaviour and state in near real-time of a physical system, as described by Kaiblinger and Woschank [92]. The authors have additionally mentioned the fact that a system such as the proposed digital system can be used to predict future states, evaluate different scenarios or parameters and ultimately be used to optimise the physical system. Although this research project has provided a first step, further research has been recommended into the usage of the proposed system in order to actually control and possibly optimise the cool chain. Along with the automatic extraction of system data, any decision-making input should likewise be fed back into the physical system. The feeding of input from the digital system towards the physical system has been classified into two stages; intermediate feedback and automatic feedback. Intermediate feedback has been formulated as the extraction of information from the digital system by a human operator which infers and or carries out any possible changes into the physical system. Automatic feedback has been considered as direct and automatic input from the digital system into the physical system. The latter is generally the goal with regards to the digitisation of processes and with respect to the presented case study would entail the connection of the digital system onto the WMS in which the business ruling has been implemented. Therefore, with regard to the six dimensions of a DT mentioned in Section 2.6, further research has been recommended into the service system, data integration and the connections between the systems.

Furthermore, since the PT model provides a virtual representation of the studied cool chain, it has been recognised at KLM Cargo that the developed PT can be utilised for what-if scenario analyses. As an example, changes in storage capacities and system configurations can be analysed through adaptations of the PT simulation model. Besides, future research could be focused on the improvement and optimisation of the decision support system algorithm, or DT in this case, in order to study to which extent the on time performance and temperature adherence can also be improved. Besides, in this research project, only temperature has been considered as an influential factor and constraint for the quality of pharmaceutical freight. Although the temperature has indeed been recognised as the major constraint in a cool chain [120], while packaging provides primary protection against humidity, light exposure may influence the quality of the freight, especially on the ramp. Therefore, also with regard to the availability of data throughout the cool chain, further research has been recommended for the utilisation of external data sources and information and the usage thereof. An example previously mentioned includes the sharing of technical details with regard to the packaging material in order to obtain virtual models of the actual freight. It has been hypothesised that information on the thermal conductivity of such packaging would allow for the modelling of the actual pharmaceutical products and their respective conditions, based on the ambient environmental conditions. However, the testing of this hypothesis has been recommended for further research in a dedicated project.

References

[1] Keith Debbage and Neil Debbage. *Air Freight Logistics*. 2021.

[2] IATA. *Air Cargo*. Accessed: 2022-05-11. 2022. URL: https://www.iata.org/en/programs/cargo/.

[3] Ben Shepherd, Anirudh Shingal, and Anasuya Raj. "Value of air cargo: Air transport and global value chains". In: *Montreal: The International Air Transport Association (IATA)* (2016).

[4] Andrea Popescu, Pinar Keskinocak, and I al Mutawaly. "The air cargo industry". In: *Intermodal transportation: Moving freight in a global economy* (2010), pp. 209–237.

[5] G.H. Hundy, A.R. Trott, and T.C. Welch. *Refrigeration and Air-Conditioning*. Jan. 2008. DOI: 10.1016/B978-0-7506-8519-1.X0001-1.

[6] Behzad Behdani, Yun Fan, and Jacqueline M Bloemhof. "Cool chain and temperature-controlled transport: An overview of concepts, challenges, and technologies". In: *Sustainable Food Supply Chains* (2019), pp. 167–183.

[7] J Gustavsson et al. "Global Food Losses and Food Waste, Study Conducted for the International Congress at Interpack 2011". In: *Food and Agriculture Organization (FAO) of the United Nations* (2011).

[8] Reiner Jedermann et al. *Reducing food losses by intelligent food logistics*. 2014.

[9] Seyed Mojtaba Hosseini Bamakan, Shima Ghasemzadeh Moghaddam, and Sajedeh Dehghan Manshadi. "Blockchain-enabled pharmaceutical cold chain: Applications, key challenges, and future trends". In: *Journal of Cleaner Production* 302 (2021), p. 127021.

[10] Yuri Yoon. "Cold Chain Management in Pharmaceutical Industry: Logistics Perspective". In: *Journal of distribution science* 12.5 (2014), pp. 33–40.

[11] Ming-Kung Yeh and Yuan-Chuan Chen. *Biopharmaceuticals*. BoD–Books on Demand, 2018.

[12] Umit Kartoglu and Julie Milstien. "Tools and approaches to ensure quality of vaccines throughout the cold chain". In: *Expert review of vaccines* 13.7 (2014), pp. 843–854.

[13] Ashvin Ashok, Michael Brison, and Yann LeTallec. "Improving cold chain systems: Challenges and solutions". In: *Vaccine* 35.17 (2017), pp. 2217–2223.

[14] Claire Sykes. "Time-and temperature-controlled transport: supply chain challenges and solutions". In: *Pharmacy and Therapeutics* 43.3 (2018), p. 154.

[15] Dewan Md Zahurul Islam and Thomas H Zunder. "The necessity for a new quality standard for freight transport and logistics in Europe". In: *European Transport Research Review* 6.4 (2014), pp. 397–410.

[16] LOG. *Monitoring and evaluation–Logistics Operational Guide (LOG)*. 2015. URL: https://dlca.logcluster.org/display/LOG/Monitoring+and+Evaluation/?msclkid=7f25ed47b5a711ec98aace0e5806b808.

[17] Thijs Defraeye et al. "Digital twins are coming: Will we need them in supply chains of fresh horticultural produce?" In: *Trends in Food Science & Technology* 109 (2021), pp. 245–258.

[18] Kanaha Shoji et al. "Mapping the postharvest life of imported fruits from packhouse to retail stores using physics-based digital twins". In: *Resources, Conservation and Recycling* 176 (2022), p. 105914.

[19] Reiner Jedermann and Walter Lang. "Wrapper Functions for Integrating Mathematical Models into Digital Twin Event Processing". In: *Sensors* 22.20 (2022), p. 7964.

[20] Thijs Defraeye et al. "Digital twins probe into food cooling and biochemical quality changes for reducing losses in refrigerated supply chains". In: *Resources, Conservation and Recycling* 149 (2019), pp. 778–794.

[21] Pieter Verboven et al. “Digital twins of food process operations: the next step for food process models?” In: *Current opinion in food science* 35 (2020), pp. 79–87.

[22] Chandrima Shrivastava et al. “Digital twins for selecting the optimal ventilated strawberry packaging based on the unique hygrothermal conditions of a shipment from farm to retailer”. In: (2022).

[23] Abderrahim Ait-Alla et al. “Simulated-based methodology for the interface configuration of cyber-physical production systems”. In: *International Journal of Production Research* 59.17 (2021), pp. 5388–5403.

[24] Franziska Kupfer et al. “The underlying drivers and future development of air cargo”. In: *Journal of Air Transport Management* 61 (2017), pp. 6–14.

[25] Rico Merkert, Eddy Van de Voorde, and Jaap de Wit. *Making or breaking-Key success factors in the air cargo market*. 2017.

[26] IATA. *Temperature Control Regulations*. 10th ed. International Air Transportation Association, Jan. 1, 2022.

[27] Jean-Paul Rodrigue. *The geography of transport systems*. Routledge, 2020.

[28] Ozan S Kumru et al. “Vaccine instability in the cold chain: mechanisms, analysis and formulation strategies”. In: *Biologicals* 42.5 (2014), pp. 237–259.

[29] John Lloyd et al. “Reducing the loss of vaccines from accidental freezing in the cold chain: the experience of continuous temperature monitoring in Tunisia”. In: *Vaccine* 33.7 (2015), pp. 902–907.

[30] Tina Comes, Kristin Bergtora Sandvik, and Bartel Van de Walle. “Cold chains, interrupted: The use of technology and information for decisions that keep humanitarian vaccines cool”. In: *Journal of Humanitarian Logistics and Supply Chain Management* (2018).

[31] BD Schoub and NA Cameron. “Problems encountered in the delivery and storage of OPV in an African country.” In: *Developments in biological standardization* 87 (1996), pp. 27–32.

[32] WHO. “Immunization supply chain and logistics”. In: *Geneva: World Health Organization* (2014).

[33] Sabeena Setia et al. “Frequency and causes of vaccine wastage”. In: *Vaccine* 20.7-8 (2002), pp. 1148–1156.

[34] Yogini Thakker and Sheila Woods. “Storage of vaccines in the community: weak link in the cold chain?” In: *British medical journal* 304.6829 (1992), pp. 756–758.

[35] Murray Trostle et al. *Immunization Essentials: A Practical Field Guide*. USAID, 1992.

[36] Stef Lemmens et al. “A review of integrated supply chain network design models: Key issues for vaccine supply chains”. In: *Chemical Engineering Research and Design* 109 (2016), pp. 366–384.

[37] Marcelo Caldeira Pedroso and Davi Nakano. “Knowledge and information flows in supply chains: A study on pharmaceutical companies”. In: *International journal of production economics* 122.1 (2009), pp. 376–384.

[38] Mahender Pal Singh. “The pharmaceutical supply chain: A diagnosis of the state-of-the-art”. PhD thesis. Massachusetts Institute of Technology, 2005.

[39] Jia-Wei Han et al. “A comprehensive review of cold chain logistics for fresh agricultural products: Current status, challenges, and future trends”. In: *Trends in Food Science & Technology* 109 (2021), pp. 536–551.

[40] Lixing Wang, SK Kwok, and WH Ip. “A radio frequency identification and sensor-based system for the transportation of food”. In: *Journal of Food Engineering* 101.1 (2010), pp. 120–129.

[41] Ronald G Askin, Yasaman Khodadadegan, and Moeed Haghnevis. “Maximizing value of perishable products by implementing RFID technology”. In: *IIE Annual Conference. Proceedings*. Institute of Industrial and Systems Engineers (IISE). 2010, p. 1.

[42] Petros S Taoukis et al. “Food cold chain management and optimization”. In: *Emerging and traditional technologies for safe, healthy and quality food*. Springer, 2016, pp. 285–309.

[43] Wentao Wu et al. “Virtual cold chain method to model the postharvest temperature history and quality evolution of fresh fruit–A case study for citrus fruit packed in a single carton”. In: *Computers and Electronics in Agriculture* 144 (2018), pp. 199–208.

[44] Wang Tingman, Zhang Jian, and Zhang Xiaoshuan. “Fish product quality evaluation based on temperature monitoring in cold chain”. In: *African Journal of Biotechnology* 9.37 (2010), pp. 6146–6151.

[45] Lixin Lu et al. “Development and application of time–temperature indicators used on food during the cold chain logistics”. In: *Packaging Technology and Science* 26 (2013), pp. 80–90.

[46] Allen Higgins et al. “Real-Time Cold Chain Mapping”. In: *Arviem. com* (2010), pp. 1–2.

[47] Bart Terpstra, Y Zhang, and AE Akçay. *Ambient Temperature Prediction of Pharmaceutical Air Freight*.

[48] Josef Oehmen et al. “System-oriented supply chain risk management”. In: *Production planning and control* 20.4 (2009), pp. 343–361.

[49] Xiaojun Wang, Puneet Tiwari, and Xu Chen. “Communicating supply chain risks and mitigation strategies: a comprehensive framework”. In: *Production Planning & Control* 28.13 (2017), pp. 1023–1036.

[50] Rameshwar Dubey et al. “Empirical investigation of data analytics capability and organizational flexibility as complements to supply chain resilience”. In: *International Journal of Production Research* 59.1 (2021), pp. 110–128.

[51] Seyedmohsen Hosseini, Dmitry Ivanov, and Alexandre Dolgui. “Review of quantitative methods for supply chain resilience analysis”. In: *Transportation Research Part E: Logistics and Transportation Review* 125 (2019), pp. 285–307.

[52] Samuel Fosso Wamba et al. “How ‘big data’can make big impact: Findings from a systematic review and a longitudinal case study”. In: *International Journal of Production Economics* 165 (2015), pp. 234–246.

[53] Samuel Fosso Wamba et al. *Transforming operations and production management using big data and business analytics: future research directions*. 2017.

[54] Gang Wang et al. “Big data analytics in logistics and supply chain management: Certain investigations for research and applications”. In: *International journal of production economics* 176 (2016), pp. 98–110.

[55] Thanos Papadopoulos et al. “The role of Big Data in explaining disaster resilience in supply chains for sustainability”. In: *Journal of Cleaner Production* 142 (2017), pp. 1108–1118.

[56] Nezih Altay et al. “Agility and resilience as antecedents of supply chain performance under moderating effects of organizational culture within the humanitarian setting: a dynamic capability view”. In: *Production Planning & Control* 29.14 (2018), pp. 1158–1174.

[57] Hendrik Haße et al. “Digital twin for real-time data processing in logistics”. In: *Artificial Intelligence and Digital Transformation in Supply Chain Management: Innovative Approaches for Supply Chains. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 27*. Berlin: epubli GmbH. 2019, pp. 4–28.

[58] Dmitry Ivanov and Alexandre Dolgui. “A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0”. In: *Production Planning & Control* 32.9 (2021), pp. 775–788.

[59] CN Verdouw et al. “A control model for object virtualization in supply chain management”. In: *Computers in industry* 68 (2015), pp. 116–131.

[60] Rudy Negenborn. “Multi Machine Coordination for Logistics, lecture 1”. https://brightspace.tudelft.nl/d2l/le/content/279431/viewContent/1891444/View. Accessed: 2022–05-10. 2021.

[61] Hyun Kang. “The prevention and handling of the missing data”. In: *Korean journal of anesthesiology* 64.5 (2013), pp. 402–406.

[62] Marina Soley-Bori. “Dealing with missing data: Key assumptions and methods for applied analysis”. In: *Boston University* 4.1 (2013), p. 19.

[63] Chiping Nieh et al. "Evaluation of imputation methods for microbial surface water quality studies". In: *Environmental Science: Processes & Impacts* 16.5 (2014), pp. 1145–1153.

[64] Yifan Zhang and Peter J Thorburn. "Handling missing data in near real-time environmental monitoring: A system and a review of selected methods". In: *Future Generation Computer Systems* 128 (2022), pp. 63–72.

[65] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*. Vol. 793. John Wiley & Sons, 2019.

[66] Lisa Ehrlinger et al. "Treating missing data in industrial data analytics". In: *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*. IEEE. 2018, pp. 148–155.

[67] Syed A Imtiaz and Sirish L Shah. "Treatment of missing values in process data analysis". In: *The Canadian Journal of Chemical Engineering* 86.5 (2008), pp. 838–858.

[68] Donald B Rubin. "Inference and missing data". In: *Biometrika* 63.3 (1976), pp. 581–592.

[69] Joseph L Schafer and John W Graham. "Missing data: our view of the state of the art." In: *Psychological methods* 7.2 (2002), p. 147.

[70] Jayson Nissen, Robin Donatello, and Ben Van Dusen. "Missing data and bias in physics education research: A case for using multiple imputation". In: *Physical Review Physics Education Research* 15.2 (2019), p. 020106.

[71] Patricia R Houck et al. "Estimating treatment effects from longitudinal clinical trial data with missing values: comparative analyses using different methods". In: *Psychiatry research* 129.2 (2004), pp. 209–215.

[72] Melissa J Azur et al. "Multiple imputation by chained equations: what is it and how does it work?" In: *International journal of methods in psychiatric research* 20.1 (2011), pp. 40–49.

[73] CHAPTER OUTLINE HEAD. "Missing Data Imputation Methods and Their Performance With Biodistance Analyses". In: (2016).

[74] Janus Christian Jakobsen et al. "When and how should multiple imputation be used for handling missing data in randomised clinical trials–a practical guide with flowcharts". In: *BMC medical research methodology* 17.1 (2017), pp. 1–10.

[75] Anders W Jørgensen et al. "Comparison of results from different imputation techniques for missing data from an anti-obesity drug trial". In: *PLoS One* 9.11 (2014), e111964.

[76] Indrė Žliobaitė and Jaakko Hollmen. "Optimizing regression models for data streams with missing values". In: *Machine learning* 99.1 (2015), pp. 47–73.

[77] Vanessa Buhrmester, David Münch, and Michael Arens. "Analysis of explainers of black box deep neural networks for computer vision: A survey". In: *Machine Learning and Knowledge Extraction* 3.4 (2021), pp. 966–989.

[78] Daniel Justus et al. "Predicting the computational cost of deep learning models". In: *2018 IEEE international conference on big data (Big Data)*. IEEE. 2018, pp. 3873–3882.

[79] Xiang Zhang et al. "Deep neural network hyperparameter optimization with orthogonal array tuning". In: *International conference on neural information processing*. Springer. 2019, pp. 287–295.

[80] Seiichi Nakajima. "Introduction to TPM: total productive maintenance." In: *Productivity Press, Inc.* (1988).

[81] T. Dunn. "OEE effectiveness". In: *Manufacturing Flexible Packaging: Materials, Machinery, and Techniques* (2014). Cited by: 4, pp. 77–85. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85042946646&partnerID=40&md5=e950049337983045397f8fef58eb2c8f.

[82] Simon Dalmolen et al. "Transportation performances measures and metrics: Overall transportation effectiveness (OTE): A framework, prototype and case study". In: *2013 46th Hawaii International Conference on System Sciences*. IEEE. 2013, pp. 4186–4195.

[83] PW Prickett. "An integrated approach to autonomous maintenance management". In: *Integrated Manufacturing Systems* (1999).

[84] SE Gouvea da Costa and E Pinheiro de Lima. “Uses and misuses of the’overall equipment effectiveness’ for production management”. In: *IEEE international engineering management conference*. Vol. 2. IEEE. 2002, pp. 816–820.

[85] Martha Patricia García. “Plug and lean: modelo de mejora continua apoyado en tecnología inalámbrica para incrementar la efectividad de las operaciones de manufactura”. PhD thesis. Universidad de Navarra, 2011.

[86] Peter Muchiri and Liliane Pintelon. “Performance measurement using overall equipment effectiveness (OEE): literature review and practical application discussion”. In: *International journal of production research* 46.13 (2008), pp. 3517–3535.

[87] Bulent Dal, Phil Tugwell, and Richard Greatbanks. “Overall equipment effectiveness as a measure of operational improvement–a practical analysis”. In: *International Journal of Operations & Production Management* 20.12 (2000), pp. 1488–1502.

[88] Marcos MO Pinto, David JK Goldberg, and João SL Cardoso. “Benchmarking operational efficiency of port terminals using the OEE indicator”. In: *Maritime Economics & Logistics* 19.3 (2017), pp. 504–517.

[89] Andrés Muñoz-Villamizar et al. “Using OEE to evaluate the effectiveness of urban freight transportation systems: A case study”. In: *International Journal of Production Economics* 197 (2018), pp. 232–242.

[90] Jesús García-Arca, J Carlos Prado-Prado, and Arturo J Fernández-González. “Integrating KPIs for improving efficiency in road transport”. In: *International Journal of Physical Distribution & Logistics Management* (2018).

[91] Obaadah Mekhallalati. “A New Method of Measuring Overall Warehouse Performance: An Automated E-Commerce Retail Warehouse”. In: (2022).

[92] Alexander Kaiblinger and Manuel Woschank. “State of the art and future directions of digital twins for production logistics: a systematic literature review”. In: *Applied Sciences* 12.2 (2022), p. 669.

[93] Averill M Law, W David Kelton, and W David Kelton. *Simulation modeling and analysis*. Vol. 3. Mcgraw-hill New York, 2007.

[94] Andrei Borshchev. *The big book of simulation modeling: multimethod modeling with AnyLogic 6*. AnyLogic North America, 2013.

[95] I Agostino et al. “Modeling and simulation of operations: A case study in a port terminal of vale S/A”. In: *New global perspectives on industrial engineering and management*. Springer, 2019, pp. 91–99.

[96] Ali Dorri, Salil S Kanhere, and Raja Jurdak. “Multi-agent systems: A survey”. In: *Ieee Access* 6 (2018), pp. 28573–28593.

[97] Shahaboddin Shamshirband et al. “An appraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique”. In: *Engineering Applications of Artificial Intelligence* 26.9 (2013), pp. 2105–2127.

[98] Franziska Klügl and Ana LC Bazzan. “Agent-based modeling and simulation”. In: *Ai Magazine* 33.3 (2012), pp. 29–29.

[99] Charles M Macal and Michael J North. “Agent-based modeling and simulation”. In: *Proceedings of the 2009 winter simulation conference (WSC)*. IEEE. 2009, pp. 86–98.

[100] Peer-Olaf Siebers et al. “Discrete-event simulation is dead, long live agent-based simulation!” In: *Journal of Simulation* 4.3 (2010), pp. 204–210.

[101] Jaap Ottjes and Hans Veeke. “Process interaction modeling and simulation”. https://brightspace.tudelft.nl/d2l/le/content/279433/viewContent/2121719/View. Accessed: 2022–10-28. 2012.

[102] James W Hooper. “Strategy-related characteristics of discrete-event languages and models”. In: *Simulation* 46.4 (1986), pp. 153–159.

[103] Su Min Jeon and Gitae Kim. “A survey of simulation modeling techniques in production planning and control (PPC)”. In: *Production Planning & Control* 27.5 (2016), pp. 360–377.

[104] Michael Schluse and Juergen Rossmann. “From simulation to experimentable digital twins: Simulation-based development and operation of complex technical systems”. In: *2016 IEEE International Symposium on Systems Engineering (ISSE)*. IEEE. 2016, pp. 1–6.

[105] Tiep Nguyen et al. “Knowledge mapping of digital twin and physical internet in Supply Chain Management: A systematic literature review”. In: *International Journal of Production Economics* 244 (2022), p. 108381.

[106] Wladimir Hofmann and Fredrik Branding. “Implementation of an IoT-and cloud-based digital twin for real-time decision support in port operations”. In: *IFAC-PapersOnLine* 52.13 (2019), pp. 2104–2109.

[107] Werner Kritzinger et al. “Digital Twin in manufacturing: A categorical literature review and classification”. In: *IFAC-PapersOnLine* 51.11 (2018), pp. 1016–1022.

[108] Fábio Coelho, Susana Relvas, and AP Barbosa-Póvoa. “Simulation-based decision support tool for in-house logistics: the basis for a digital twin”. In: *Computers & Industrial Engineering* 153 (2021), p. 107094.

[109] Fei Tao and Meng Zhang. “Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing”. In: *Ieee Access* 5 (2017), pp. 20418–20427.

[110] Fei Tao et al. “Digital twin in industry: State-of-the-art”. In: *IEEE Transactions on Industrial Informatics* 15.4 (2018), pp. 2405–2415.

[111] Cor Verdouw et al. “Digital twins in smart farming”. In: *Agricultural Systems* 189 (2021), p. 103046.

[112] Jannicke Baalsrud Hauge et al. “Digital twin testbed and practical applications in production logistics with real-time location data”. In: *International Journal of Industrial Engineering and Management* 12.2 (2021), p. 129.

[113] Mihai Stan, Theodor Borangiu, and Silviu Răileanu. “Data-and model-driven digital twins for design and logistics control of product distribution”. In: *2021 23rd International Conference on Control Systems and Computer Science (CSCS)*. IEEE. 2021, pp. 33–40.

[114] David Hensens. “Improving the Overall Performance of the Warehouse Processes of Temperature-Sensitive Goods: Within the Warehouse of KLM Cargo at the Schiphol hub”. In: (2019).

[115] Utomo Pujianto, Aji Prasetya Wibawa, Muhammad Iqbal Akbar, et al. “K-nearest neighbor (k-NN) based missing data imputation”. In: *2019 5th International Conference on Science in Information Technology (ICSITech)*. IEEE. 2019, pp. 83–88.

[116] Zhexue Huang. “Extensions to the k-means algorithm for clustering large data sets with categorical values”. In: *Data mining and knowledge discovery* 2.3 (1998), pp. 283–304.

[117] Zhaobin Liu, Satish Sharma, and Sandeep Datla. “Imputation of missing traffic data during holiday periods”. In: *Transportation Planning and Technology* 31.5 (2008), pp. 525–544.

[118] Lorenzo Beretta and Alessandro Santaniello. “Nearest neighbor imputation algorithms: a critical evaluation”. In: *BMC medical informatics and decision making* 16.3 (2016), pp. 197–208.

[119] Robert G Sargent. “Model verification and validation”. In: *Modeling and simulation in the systems engineering life cycle*. Springer, 2015, pp. 57–65.

[120] Gordan Badurina, Zvonimir Majić, and Stanislav Pavlin. “Evaluation of air transportation under controlled room temperature for pharmaceuticals”. In: *Promet-Traffic&Transportation* 23.2 (2011), pp. 121–130.

A

Scientific research paper

On the following pages, a summary of this research report has been presented in the form of a scientific research paper.

Towards the development of a Digital Twin for the improvement of cool chain operational quality: A KLM Cargo Case Study

J. A. Sijtsma *, Dr. ir. Y. Pang †, Prof. dr. R. R. Negenborn †, B. Krol ‡

Abstract—The demand for accurate and effective cool chains has been expected to increase, especially for pharmaceuticals in the air freight industry. However, several problems and challenges remain such as cool chain breaks and cool storage capacity constraints while there are rarely routine systems in place for consistent insight into the operational quality of such systems. Besides, the concept of a DT has received increasing attention in the literature, while a multitude of applications have been found in fresh cool chains. Therefore, the DT concept has been applied to a pharmaceutical cool chain for operational quality improvement. Firstly, a novel operational quality metric has been proposed: the OCCE. Consequently, a cool chain at KLM Cargo has been studied and modelled by means of the DES technique in order to derive a virtual representation. Consequently, the DT concept has been applied through the implementation of a decision support module for cool storage decision-making. The model implementation has shown that the cool chain operational quality has been improved while the average exposure of freight has decreased.

Index Terms—Digital Twin, Cool Chain, OEE, DES, Decision Support, Decision-making, Monitoring, Operational Quality Management

I. INTRODUCTION

A. Research background

THE air freight logistics industry has become an increasingly important part of the modern global economy [1]. Annually, airlines transport over 52 million metric tons of cargo with a value equivalent to $6.8 trillion [2]. Even though air freight shipments may account for less than 1% of global trade by volume, the total value accounts for 35% of all global shipments [3]. A demand increase has been anticipated for certain specialist products such as pharmaceuticals, cut flowers and medical diagnostic devices [1], which are distributed through the so-called cool chain. A cool chain includes all steps and facilities for storing, handling and transportation of perishable products, for which controlled temperature conditions must be maintained [4] and is thus also aimed at preserving the quality of products throughout the chain [5]. Especially pharmaceutical shipments impose significant risks in terms of consumer health whenever product quality has been affected throughout the cool chain [6]. Furthermore, the reliance on and demand for a temperature-controlled cool chain for the pharmaceutical industry is actively driven by the increasing sensitivity of pharmaceutical products to environmental conditions [7], [8]. In general, cool chain management is critical due to the high costs and significant shipment values of pharmaceuticals. However, several problems and challenges remain. Firstly, especially in the air freight industry, a network may contain many handovers of shipments that in principle constitute to breaks in the cool chain during which the risk of temperature excursions significantly increases. Furthermore, there are rarely routine systems in place to provide consistent insight into cool chain performance and enable day-to-day performance management [9].

B. Problem definition

One actor within the air freight industry facing cool chain challenges is KLM Cargo. With tightening regulations [10] and a decreasing market share, the urgency for operational quality improvement has arisen. In essence, the operational quality of a cool chain has been defined as the performance of the system with respect to certain criteria or service levels, in line with the definition for transport logistics quality [11]. In other words, operational quality has been understood as the effectiveness of the system. A contributing factor which has a significant impact on the operational quality is capacity constrained cool storage. Furthermore, decision-making is usually performed through static business ruling while there are no insights into the state of the freight and the cool chain as a whole. Therefore, improved decision-making may provide an opportunity to improve the cool chain with the given infrastructure and thus the problem definition has been stated as follows:

“Currently, there is no capability in place for real-time decision-making in order to improve the cool chain operational quality based on the actual system state with the existing infrastructure.”

C. Research objective

Based on the described problem definition, the goal of the research has been formulated as follows:

”The research project has been aimed at the development of a real-time decision-making methodology in order to improve the operational quality of the pharmaceutical cool chain.”

*MSc. Multi-Machine Engineering - Delft University of Technology
†Faculty of 3mE — Marine and Transport Technology
‡Air France KLM Martinair Cargo

The goal of decision-making improvement entails the aim of utilising process and environmental data in order to determine, support and possibly improve the decision-making with regard to the cool storage of pharmaceutical freight. On the one hand, cool storage provides a benefit to the handled freight with regard to the environmental conditions and cool chain breaks. On the other hand, cool storage facilities are typically constrained with regard to capacity. Therefore, this paper has been aimed at quantifying the potential improvement of cool chains, by answering the following research question:

"To what extent can the pharmaceutical cool chain operational quality be improved through the development of a real-time decision-making methodology?"

D. Scope

In order to ensure the feasibility of the performed research, the scope with respect to the performed case study has been limited to warehouse handling at KLM Cargo. Besides, only pharmaceutical shipments have been considered which are handled in a truck-to-aircraft transit flow. Finally, with regard to the DT concept, automatic data connections to and from the physical system have not been realised.

E. Methodology

In order to answer the main research question and obtain the research goal, several methodologies have been used. Firstly, the Overall Equipment Effectiveness (OEE) methodology has been adapted into a novel effectiveness metric for the operational quality of a cool chain. Secondly, for the frequently occurring issue of missing data in industry, k-Nearest-Neighbour (kNN) data imputation has been utilised in order to deal with missing data entries. Then, a manually programmed Discrete Event Simulation (DES) model has been used in order to obtain a virtual representation of the studied system. DES has been chosen since it is the most used and state-of-the-art technique for logistics systems simulation [12]. Finally, the DT concept has been adopted in order to develop a real-time decision-making methodology for pharmaceutical cool chains since it has received little attention in the literature, in contrast to fresh cool chain studies [13]–[18]. In specific, a DT development methodology introduced by [19] has been used as the basis for the development of the improvement method. Consequently, the contribution of this research project has been recognised as an extension of the application of the DT concept into the domain of the pharmaceutical air freight cool chain in order to improve the operational quality by means of real-time decision-making.

II. STATE OF THE ART

A. Cool chain management

A wide range of cool chain disruptions and challenges can be encountered, which generally can be associated with three failure categories [20]. Firstly, a disruption of material flow is typically attributed to a lack of infrastructure or equipment, or the failure thereof. Furthermore, a lack of the ability to extract useful information from the cool chain may lead to information gaps, which increase the difficulty to assess the performance of the system. Finally, there may in general be a failure of decision making or lack of operational support available. This could be attributed to both a lack of information available or the inability to utilise available data for improved decision making. Improving capacity is generally not a viable option for any short term improvements since apart from significant investments, the lead time and procurement of cool chain equipment may take up to two years [9]. Nonetheless, capacity constraints have been recognised as a typically encountered problem in practice [21], [22]. Therefore, regarding the information and decisions layers, it has been noted that one of the main research directions is to ensure the integrity of the cool chain and its precise control [23]. Significant efforts have been spent on the improvement of cool chains, especially in the information and decisions domain. For example, attempts have been made to model and improve shelf life based decision-making in fresh cool chains [24]–[29]. However, such methods of improvement are not easily transferred to the pharmaceutical cool chain given the limited information sharing on packaging and product characteristics between cool chain partners. In general, cool chain managers are interested in decision-making support in order to monitor and recognise disruptions in real time while being able to determine the required actions to deal with such situations [30]–[33]. Some researches have pointed out a trend in supply chain management towards a Digital Twin (DT), i.e. computerised models which represent a physical object in real time [34]–[38]. It has been recognised that a DT can offer considerable potential especially in logistics [39], [40] and virtualisation within supply chains is an important topic in research [41].

B. Performance evaluation

In order to assess the operational quality of a system such as a cool chain, certain performance indicators are required. The Overall Equipment Effectiveness (OEE) indicator has been introduced within the Total Productive Maintenance conceptual framework [42]. In principle, OEE is a metric which can be used to measure the effectiveness of production equipment and how effectively a manufacturing operation is realised [43]. The original definition of OEE has been expressed as a percentage resulting from the multiplication of three rates:

$$\text{OEE} = \text{Availability} \cdot \text{Performance} \cdot \text{Quality} \tag{1}$$

where each measure has been defined as:

- Availability: the actual time used versus the planned time;
- Performance: the actual production versus the standard during the actual time used in production;

- Quality: the number of faulty products produced in comparison to the total number produced.

Although the OEE is computed in percentages, time is the central metric unit for the respective sub-measures [44]. Besides production systems, the OEE methodology has been applied to other fields as well [45]–[47] while it has been found to be applied in a multitude of DT studies [48]. Therefore, the OEE method has similarly been adopted into an effectiveness metric for a cool chain: the Overall Cool Chain Effectiveness (OCCE), which has been defined as follows:

$$\mathrm{OCCE} = R_A \cdot R_{OTP} \cdot R_{TA} \tag{2}$$

where R_A is the cool storage availability rate, R_{OTP} is the on time performance rate and R_{TA} the temperature adherence rate. Each rate has been further elaborated on:

- Cool storage availability: this rate refers to the utilisation of cool storage facilities. Since the unavailability of cool storage facilities is common, the availability rate indicates to which extent a cool chain provides availability of required infrastructure.
- On time performance: in principle, the operational quality of a cool chain can be seen as twofold; on the one hand timeliness and on the other the extent to which freight is handled according to the required environmental conditions. The on time performance provides an indication to which extent the timeliness of a cool chain is as per requested.
- Temperature adherence: this rate provides an indication to which extent freight is handled according to environmental specification by the shipper.

C. Digital Twin

The definition of a DT has been specified as a virtual representation of a real world subject or a real world object which contains models of its data, functionality and communication interfaces [49]. Besides an essential characteristic of a DT has been noted as the capability to generate virtual instances and control the changes of a physical object in real-time [50]. The definition has been extended by considering the exploitation of real-time synchronisation of data [51]. Three stages of DT integration have been distinguished based on the degree of automation of data flows [52], seen in Figure 1.

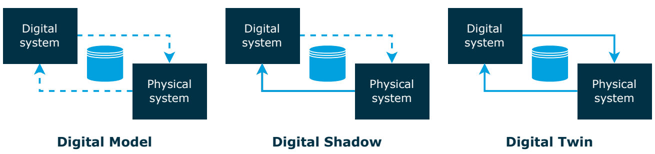


Fig. 1. The three stages of Digital Twin integration depending on the automation of data flow, adapted from [51]

Therefore, a DT may then be considered as containing an automatic data flow to and from the physical system. Besides, the assumption that a DT should add additional functionality besides a virtual representation has been stated [19], which has been summarised in a development methodology seen in Figure 2.

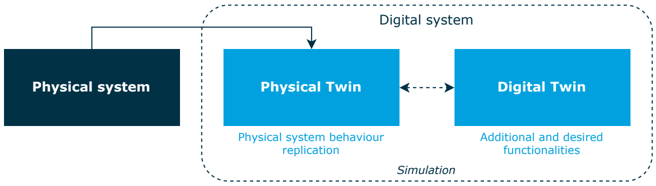


Fig. 2. DT development study approach, adapted from [19]

The physical system in this case is comprised of the cool chain, which is represented in a simulation model by the Physical Twin (PT) while the DT offers additional functionalities through for instance decision-making support. Therefore, a DT has been defined in this work as the additional functionalities offered by the digital system through interactions with the PT model representation of the physical system, while utilising automatic data connections. In the literature, the concept of a DT has received increasingly received attention, where the main focus has been on manufacturing while more research has been suggested for the application in logistics [39]. Several benefits have been noted in supply chains [19], [53], while the concept has been applied in different fields [52], [54]–[56] as well as fresh cool chains.

III. CASE STUDY DESCRIPTION

In order to apply the DT concept for improved decision-making in a cool chain, a case study at KLM Cargo has been performed. The studied system encompasses a pharmaceutical air freight cool chain with handling facilities located at the Schiphol hub. The facility handles pharmaceutical freight classified according to three Special Handling Codes (SHC): COL for 2 to 8 °C, CRT for 15 to 25 °C and ERT for 2 to 25 °C.

A. Process description

The studied process at KLM Cargo is comprised of a truck-to-aircraft pharmaceutical freight flow, consisting out of the so-called Through-ULD (T-ULD). T-ULDs are Unit Load Devices (ULDs) which have already been built up with freight and are thus ready for flight upon arrival at the warehouse. Therefore, the processing steps for T-ULDs mainly consist out of temporary storage and transportation on the airport premises. An overview of the studied process has been shown in Fig. Figure 3. The units flowing through the system are the T-ULDs which may contain a single Air WayBill (AWB), or shipment, or may share the ULD with other AWBs bound for the same destination. Besides, a single AWB may be comprised of multiple T-ULDs. The process can be roughly divided in arrival, Pallet Container Handling System (PCHS) handling, cool storage and ramp ride and loading. Upon arrival of a truck, it docks at the warehouse where the T-ULDs are offloaded by means of the Moving Truck Door

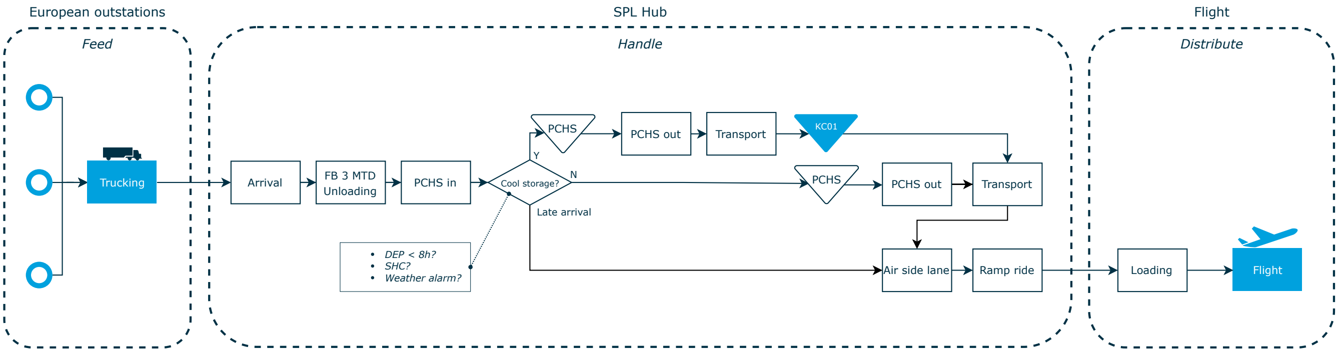


Fig. 3. Process map for transit outbound T-ULD handling at the KLM Cargo hub

(MTD). Consequently, the ULDs enter the PCHS, which is an automated storage and retrieval system. Based on static business rules, it is then determined by the Warehouse Management System (WMS) whether the T-ULD should remain in the PCHS or receive cool storage. The following business rules are in use:

- DEP < 8: any shipment with a transit time less than eight hours is not placed in cool storage.
- SHC: if the transit time exceeds eight hours, only COL an CRT T-ULDs receive cool storage.
- Weather alarm: in the case that a weather alarm is issued when the ambient temperature exceeds 18 °C or is below 5 °C, shipments with SHC ERT are stored in the CRT cool storage when the transit time exceeds eight hours.

The cool storage facility, also named KC01, contains two rooms with a set-point of 5 °C and 20 °C for COL and CRT respectively. Through a data analysis, the capacity of KC01 has been assumed at 6 T-ULDs for the COL room and 28 T-ULDs for the CRT room for the specified research scope. With respect to cool chain breaks, cool storage is usually the best option. However, the capacity of KC01 is significantly constrained, necessitating the need for the static cool storage business ruling. With regard to removal from storage, the standard removal time before Scheduled Time of Departure (STD) from the PCHS is five hours and from KC01 three hours. After removal, T-ULDs are placed in a buffer outside before being transported towards the aircraft.

B. Performance management

Throughout the studied system, data is generated, collected and used to determine the system performance by means of Key Performance Indicators (KPIs). In terms of general handling, the following deadlines have been defined:

- Handling deadline 1: T-ULD received into the MTD process 360 minutes before STD.
- Handling deadline 2: T-ULD handed over from MTD to Transport 360 minutes before STD.
- Handling deadline 3: T-ULD handed over from Transport to Ground Services 80 minutes before STD.

Similarly, the following cool chain deadlines have been defined:

- Cool chain deadline 1: a T-ULD enters the PCHS within 120 minutes after arrival.
- Cool chain deadline 2: T-ULDs with a transit time of more than or equal to eight hours enter KC01 within 180 minutes after arrival.
- Cool chain deadline 3: a T-ULD stored in KC01 is removed no longer than 180 minutes before STD.

Furthermore, the timeliness of the system is measured according to the Flown As Planned (FAP) KPI, which indicates whether a ULD has been flown on its booked flight. The temperature aspect is currently measured by the Time Out of Refrigeration (TOR), which is a measure of the total time spent at the hub versus the time spent in cool storage. Consequently, in the studied system, cool storage capacity issues have been encountered along with a lack of an overall cool chain effectiveness metric which incorporates the temperature exposure of T-ULDs.

IV. MODELLING

An overview of the proposed conceptual model has been shown in Figure 4. The model consists of the studied physical system along with a proposed digital system consisting out of the PT DES model and DT decision support module.

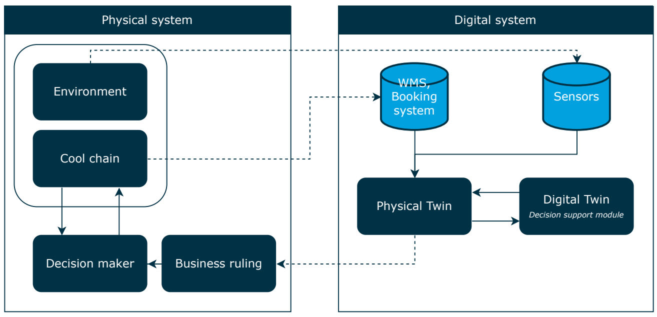


Fig. 4. Conceptual research model

The mathematical symbols which have been used throughout the remainder of this paper have been summarised below:

C_{COL}	Total capacity of the KC01 COL storage room
C_{CRT}	Total capacity of the KC01 CRT storage room
I	The set of availability measurements
n	Number of simulation runs
q_{col}	Current quantity of stored ULDs in the KC01 COL storage room
q_{crt}	Current quantity of stored ULDs in the KC01 CRT storage room
R_A	Average cool storage availability of the KC01 storage facility
$R_{A,COL}$	Cool storage availability rate of the KC01 COL storage room
$R_{A,CRT}$	Cool storage availability rate of the KC01 CRT storage room
R_{KC01}	Standard KC01 storage removal time rule
R_{OTP}	On time performance rate
R_{PCHS}	Standard PCHS storage removal time rule
R_{TA}	Temperature adherence rate
R_{TT}	Cool storage transit time business rule
S	Total quantity of handled AWBs
s_e	Number of AWBs with all ULDs having an exposure less than eight hours
s_{mf}	Quantity of AWBs with a ULD that missed the flight

A. Model development

Through the collection, handling and analysis of process data covering a period from 01-01-2021 until 01-01-2022, two input data sets for the PT model have been obtained: a T-ULD input data set and a temperature input data set. Furthermore, the processing time distributions have been derived for the development of the PT DES model representation of the studied system, which has been manually programmed in Python. Therefore, through the input data set, the flow of T-ULDs through the process has been simulated, including real-world temperature data in the various parts of the system. The OCCE metric has been applied for the PT model output, including the following rates:

$$R_{A,COL} = \frac{C_{COL} - q_{col}}{C_{COL}} \tag{3}$$

$$R_{A,CRT} = \frac{C_{CRT} - q_{crt}}{C_{CRT}} \tag{4}$$

$$R_A = \frac{\frac{\sum_{i=0}^{I} R_{A,COL}}{|I|} + \frac{\sum_{i=0}^{I} R_{A,CRT}}{|I|}}{2} \tag{5}$$

$$R_{OTP} = 1 - \frac{s_{mf}}{S} \tag{6}$$

$$R_{TA} = \frac{s_e}{S} \tag{7}$$

Exposure in this case has been noted as the cumulative time during which a T-ULD is situated in an environment outside of its respective allowable temperature range, where the maximum allowable exposure per ULD has been set at 8 hours at KLM Cargo.

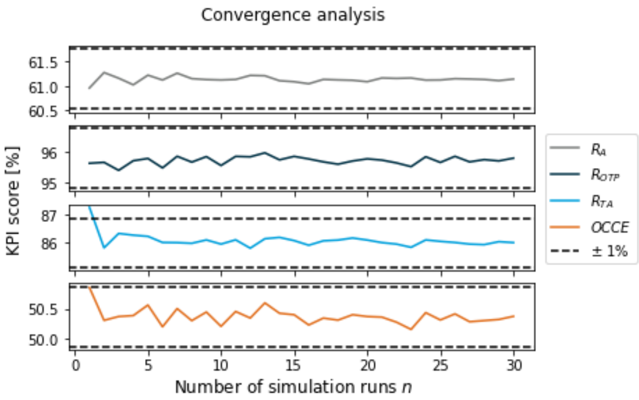


Fig. 5. Physical Twin model convergence analysis for a total of $n = 30$ runs

B. Verification and validation

A model convergence analysis has been performed in order to study the convergence of the PT model output, as seen in Figure 5. From the convergence analysis, $n = 20$ has been deemed acceptable with respect to the computational time required for $n > 20$. In order to determine whether the PT model is right and if it is the right model, verification and validation have respectively iteratively been performed [57]. The model has been concluded to be verified, while only partial validation has been obtained by means of historical performance data.

C. Model implementation

The DT concept has been manually programmed in Python as a separate module which has been loaded into the PT model, forming the digital system as seen in Figure 6. From the PT model, relevant information such as cool storage availability, T-ULD exposure, temperature in the system and the STD. This information is then utilised in the DT decision support module in order to determine real-time whether the current ULD needs cool storage

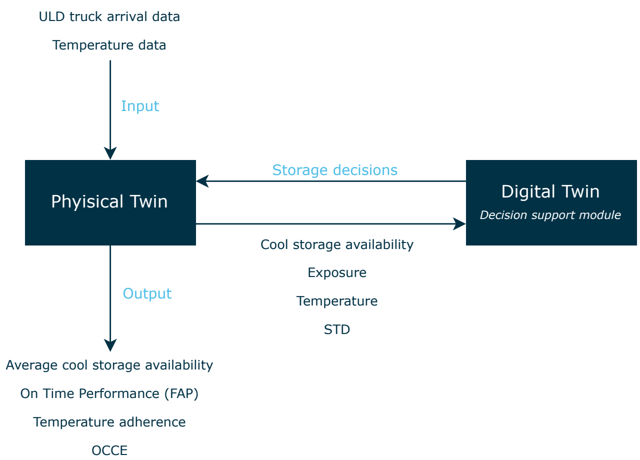


Fig. 6. Schematic overview of the proposed digital system

TABLE I
Model output results for the full simulation duration averaged over $n = 20$ runs

KPI	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [hh:mm:ss]
Baseline	54.70	74.66	64.68	95.68	85.95	53.19	04:10:41
DT implementation	55.21	80.34	67.87	95.04	85.59	55.21	04:05:32
Difference	+0.93%	+7.61%	+4.93%	-0.67%	-0.42%	+3.80%	-2.05%

TABLE II
Model output differences compared to the baseline scenario for the different periods throughout the year

	$R_{A,COL}$ [%]	$R_{A,CRT}$ [%]	R_A [%]	R_{OTP} [%]	R_{TA} [%]	OCCE [%]	Average exposure [%]
General	+0.93	+7.61	+4.93	-0.67	-0.42	+3.80	-2.05
Winter	+1.50	+1.41	+1.09	-0.76	-0.08	+0.24	-0.76
Spring - Summer	+1.49	+11.74	+7.33	-0.67	-0.21	+6.39	-1.71
Autumn - Winter	+0.67	+3.37	+2.21	-0.83	-0.02	+1.26	-2.35

based on the expected exposure in the cool chain. In the case that the expected exposure for PCHS storage is less than or equal to the exposure with KC01 cool storage, then the decision to store in the PCHS is fed back to the PT model. In the case that a T-ULD should receive cool storage and there is insufficient availability, the DT module determines whether to remove a ULD based on expected exposure in order to make space. The resulting changes on the PT model during a simulation run are reflected in the OCCE KPIs which have been compared to the baseline case study system.

V. Results

The resulting model output for the baseline scenario and implemented DT module has been summarised in Table I for a total of $n = 20$ runs over the whole simulation duration of one year. Furthermore, the results for different periods of interest during the year have been extracted in a similar way, which has been summarised in Table II. In general, it has been observed that the OCCE and thus operational quality of the cool chain has improved by 3.80% through the implemented DT concept. The improvement has largely been obtained through the improvement of the CRT cool storage availability. Besides, the average T-ULD exposure has been reduced by 2.05%. Considering the different periods, it has been observed that the CRT availability increase is most prominent during spring and summer, which is typically a critical period. Furthermore, the on time performance and temperature adherence have slightly decreased, which has been attributed to the variance observed of these rates in the convergence analysis for $n > 20$.

VI. Conclusion and Recommendations

In conclusion, this paper has introduced a novel metric for the effectiveness of a cool chain as a whole. Furthermore, the application of the DT concept has been extended to a pharmaceutical cool chain through a case study at KLM Cargo. It has been found that through the implementation of the proposed digital system including a DT decision-support module has been able to improve the operational quality of the cool chain as quantified by the OCCE. Therefore, it has been deemed possible to instate a quantifiable operational quality improvement in pharmaceutical cool chains through the joint implementation of the OCCE metric with the DT concept for real-time decision-making.

Although it has been found that the proposed improvement method by means of the digital system is able to improve the cool chain operational quality, the actual implementation into a cool chain should also be considered. In general, further research has been recommended with regards to the integration and or connection of the proposed digital system with the systems in use in the respective cool chain such as the WMS and temperature sensor networks. Besides, the connection to the physical system should likewise be considered for future research. Furthermore, the improvement and optimisation of the decision support system algorithm can be studied in order to determine to which extent the on time performance and temperature adherence can also be improved. With regard to the availability of data throughout the cool chain, further research has been recommended for the utilisation of external data sources and information and the usage thereof. It has been hypothesised that information on the thermal conductivity of pharmaceutical freight packaging would allow for the modelling of the actual pharmaceutical products and their respective conditions, based on the ambient environmental conditions. However, the testing of this hypothesis has been recommended for further research in a dedicated project.

References

[1] K. Debbage and N. Debbage, Air Freight Logistics, 2021.

[2] IATA. (2022) Air cargo. Accessed: 2022-05-11. [Online]. Available: https://www.iata.org/en/programs/cargo/

[3] B. Shepherd, A. Shingal, and A. Raj, “Value of air cargo: Air transport and global value chains,” Montreal: The International Air Transport Association (IATA), 2016.

[4] G. Hundy, A. Trott, and T. Welch, Refrigeration and Air-Conditioning, 01 2008.

[5] B. Behdani, Y. Fan, and J. M. Bloemhof, “Cool chain and temperature-controlled transport: An overview of concepts, challenges, and technologies,” Sustainable Food Supply Chains, pp. 167–183, 2019.

[6] S. M. H. Bamakan, S. G. Moghaddam, and S. D. Manshadi, "Blockchain-enabled pharmaceutical cold chain: Applications, key challenges, and future trends," Journal of Cleaner Production, vol. 302, p. 127021, 2021.
[7] Y. Yoon, "Cold chain management in pharmaceutical industry: Logistics perspective," Journal of distribution science, vol. 12, no. 5, pp. 33–40, 2014.
[8] M.-K. Yeh and Y.-C. Chen, Biopharmaceuticals. BoD–Books on Demand, 2018.
[9] A. Ashok, M. Brison, and Y. LeTallec, "Improving cold chain systems: Challenges and solutions," Vaccine, vol. 35, no. 17, pp. 2217–2223, 2017.
[10] C. Sykes, "Time-and temperature-controlled transport: supply chain challenges and solutions," Pharmacy and Therapeutics, vol. 43, no. 3, p. 154, 2018.
[11] D. M. Z. Islam and T. H. Zunder, "The necessity for a new quality standard for freight transport and logistics in europe," European Transport Research Review, vol. 6, no. 4, pp. 397–410, 2014.
[12] S. M. Jeon and G. Kim, "A survey of simulation modeling techniques in production planning and control (ppc)," Production Planning & Control, vol. 27, no. 5, pp. 360–377, 2016.
[13] T. Defraeye, C. Shrivastava, T. Berry, P. Verboven, D. Onwude, S. Schudel, A. Bühlmann, P. Cronje, and R. M. Rossi, "Digital twins are coming: Will we need them in supply chains of fresh horticultural produce?" Trends in Food Science & Technology, vol. 109, pp. 245–258, 2021.
[14] K. Shoji, S. Schudel, D. Onwude, C. Shrivastava, and T. Defraeye, "Mapping the postharvest life of imported fruits from packhouse to retail stores using physics-based digital twins," Resources, Conservation and Recycling, vol. 176, p. 105914, 2022.
[15] R. Jedermann and W. Lang, "Wrapper functions for integrating mathematical models into digital twin event processing," Sensors, vol. 22, no. 20, p. 7964, 2022.
[16] T. Defraeye, G. Tagliavini, W. Wu, K. Prawiranto, S. Schudel, M. A. Kerisima, P. Verboven, and A. Bühlmann, "Digital twins probe into food cooling and biochemical quality changes for reducing losses in refrigerated supply chains," Resources, Conservation and Recycling, vol. 149, pp. 778–794, 2019.
[17] P. Verboven, T. Defraeye, A. K. Datta, and B. Nicolai, "Digital twins of food process operations: the next step for food process models?" Current opinion in food science, vol. 35, pp. 79–87, 2020.
[18] C. Shrivastava, S. Schudel, K. Shoji, D. Onwude, F. P. da Silva, D. Turan, M. Paillart, and T. Defraeye, "Digital twins for selecting the optimal ventilated strawberry packaging based on the unique hygrothermal conditions of a shipment from farm to retailer," 2022.
[19] A. Ait-Alla, M. Kreutz, D. Rippel, M. Lütjen, and M. Freitag, "Simulated-based methodology for the interface configuration of cyber-physical production systems," International Journal of Production Research, vol. 59, no. 17, pp. 5388–5403, 2021.
[20] T. Comes, K. B. Sandvik, and B. Van de Walle, "Cold chains, interrupted: The use of technology and information for decisions that keep humanitarian vaccines cool," Journal of Humanitarian Logistics and Supply Chain Management, 2018.
[21] M. C. Pedroso and D. Nakano, "Knowledge and information flows in supply chains: A study on pharmaceutical companies," International journal of production economics, vol. 122, no. 1, pp. 376–384, 2009.
[22] M. P. Singh, "The pharmaceutical supply chain: A diagnosis of the state-of-the-art," Ph.D. dissertation, Massachusetts Institute of Technology, 2005.
[23] J.-W. Han, M. Zuo, W.-Y. Zhu, J.-H. Zuo, E.-L. Lü, and X.-T. Yang, "A comprehensive review of cold chain logistics for fresh agricultural products: Current status, challenges, and future trends," Trends in Food Science & Technology, vol. 109, pp. 536–551, 2021.
[24] L. Wang, S. Kwok, and W. Ip, "A radio frequency identification and sensor-based system for the transportation of food," Journal of Food Engineering, vol. 101, no. 1, pp. 120–129, 2010.
[25] R. G. Askin, Y. Khodadadegan, and M. Haghnevis, "Maximizing value of perishable products by implementing rfid technology," in IIE Annual Conference. Proceedings. Institute of Industrial and Systems Engineers (IISE), 2010, p. 1.
[26] P. S. Taoukis, E. Gogou, T. Tsironi, M. Giannoglou, E. Dermesonlouoglou, and G. Katsaros, "Food cold chain management and optimization," in Emerging and traditional technologies for safe, healthy and quality food. Springer, 2016, pp. 285–309.
[27] W. Wu, P. Cronjé, B. Nicolai, P. Verboven, U. L. Opara, and T. Defraeye, "Virtual cold chain method to model the postharvest temperature history and quality evolution of fresh fruit–a case study for citrus fruit packed in a single carton," Computers and Electronics in Agriculture, vol. 144, pp. 199–208, 2018.
[28] W. Tingman, Z. Jian, and Z. Xiaoshuan, "Fish product quality evaluation based on temperature monitoring in cold chain," African Journal of Biotechnology, vol. 9, no. 37, pp. 6146–6151, 2010.
[29] L. Lu, W. Zheng, Z. Lv, and Y. Tang, "Development and application of time–temperature indicators used on food during the cold chain logistics," Packaging Technology and Science, vol. 26, pp. 80–90, 2013.
[30] J. Oehmen, A. Ziegenbein, R. Alard, and P. Schönsleben, "System-oriented supply chain risk management," Production planning and control, vol. 20, no. 4, pp. 343–361, 2009.
[31] X. Wang, P. Tiwari, and X. Chen, "Communicating supply chain risks and mitigation strategies: a comprehensive framework," Production Planning & Control, vol. 28, no. 13, pp. 1023–1036, 2017.
[32] R. Dubey, A. Gunasekaran, S. J. Childe, S. Fosso Wamba, D. Roubaud, and C. Foropon, "Empirical investigation of data analytics capability and organizational flexibility as complements to supply chain resilience," International Journal of Production Research, vol. 59, no. 1, pp. 110–128, 2021.
[33] S. Hosseini, D. Ivanov, and A. Dolgui, "Review of quantitative methods for supply chain resilience analysis," Transportation Research Part E: Logistics and Transportation Review, vol. 125, pp. 285–307, 2019.
[34] S. F. Wamba, S. Akter, A. Edwards, G. Chopin, and D. Gnanzou, "How 'big data'can make big impact: Findings from a systematic review and a longitudinal case study," International Journal of Production Economics, vol. 165, pp. 234–246, 2015.
[35] S. F. Wamba, E. W. Ngai, F. Riggins, and S. Akter, "Transforming operations and production management using big data and business analytics: future research directions," pp. 2–9, 2017.
[36] G. Wang, A. Gunasekaran, E. W. Ngai, and T. Papadopoulos, "Big data analytics in logistics and supply chain management: Certain investigations for research and applications," International journal of production economics, vol. 176, pp. 98–110, 2016.
[37] T. Papadopoulos, A. Gunasekaran, R. Dubey, N. Altay, S. J. Childe, and S. Fosso-Wamba, "The role of big data in explaining disaster resilience in supply chains for sustainability," Journal of Cleaner Production, vol. 142, pp. 1108–1118, 2017.
[38] N. Altay, A. Gunasekaran, R. Dubey, and S. J. Childe, "Agility and resilience as antecedents of supply chain performance under moderating effects of organizational culture within the humanitarian setting: a dynamic capability view," Production Planning & Control, vol. 29, no. 14, pp. 1158–1174, 2018.
[39] H. Haße, B. Li, N. Weißenberg, J. Cirullies, and B. Otto, "Digital twin for real-time data processing in logistics," in Artificial Intelligence and Digital Transformation in Supply Chain Management: Innovative Approaches for Supply Chains. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 27. Berlin: epubli GmbH, 2019, pp. 4–28.
[40] D. Ivanov and A. Dolgui, "A digital supply chain twin for managing the disruption risks and resilience in the era of industry 4.0," Production Planning & Control, vol. 32, no. 9, pp. 775–788, 2021.
[41] C. Verdouw, A. J. Beulens, H. A. Reijers, and J. G. van der Vorst, "A control model for object virtualization in supply chain management," Computers in industry, vol. 68, pp. 116–131, 2015.
[42] S. Nakajima, "Introduction to tpm: total productive maintenance." Productivity Press, Inc., 1988.
[43] T. Dunn, "Oee effectiveness," Manufacturing Flexible Packaging: Materials, Machinery, and Techniques, p. 77 – 85, 2014, cited by: 4. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85042946646&partnerID=40&md5=e950049337983045397f8fef58eb2c8f

[44] S. Dalmolen, H. Moonen, I. Iankoulova, and J. van Hillegersberg, "Transportation performances measures and metrics: Overall transportation effectiveness (ote): A framework, prototype and case study," in 2013 46th Hawaii International Conference on System Sciences. IEEE, 2013, pp. 4186–4195.
[45] M. M. Pinto, D. J. Goldberg, and J. S. Cardoso, "Benchmarking operational efficiency of port terminals using the oee indicator," Maritime Economics & Logistics, vol. 19, no. 3, pp. 504–517, 2017.
[46] A. Muñoz-Villamizar, J. Santos, J. R. Montoya-Torres, and C. Jaca, "Using oee to evaluate the effectiveness of urban freight transportation systems: A case study," International Journal of Production Economics, vol. 197, pp. 232–242, 2018.
[47] J. García-Arca, J. C. Prado-Prado, and A. J. Fernández-González, "Integrating kpis for improving efficiency in road transport," International Journal of Physical Distribution & Logistics Management, 2018.
[48] A. Kaiblinger and M. Woschank, "State of the art and future directions of digital twins for production logistics: a systematic literature review," Applied Sciences, vol. 12, no. 2, p. 669, 2022.
[49] M. Schluse and J. Rossmann, "From simulation to experimentable digital twins: Simulation-based development and operation of complex technical systems," in 2016 IEEE International Symposium on Systems Engineering (ISSE). IEEE, 2016, pp. 1–6.
[50] T. Nguyen, Q. H. Duong, T. Van Nguyen, Y. Zhu, and L. Zhou, "Knowledge mapping of digital twin and physical internet in supply chain management: A systematic literature review," International Journal of Production Economics, vol. 244, p. 108381, 2022.
[51] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," IFAC-PapersOnLine, vol. 51, no. 11, pp. 1016–1022, 2018.
[52] W. Hofmann and F. Branding, "Implementation of an iot-and cloud-based digital twin for real-time decision support in port operations," IFAC-PapersOnLine, vol. 52, no. 13, pp. 2104–2109, 2019.
[53] F. Tao, H. Zhang, A. Liu, and A. Y. Nee, "Digital twin in industry: State-of-the-art," IEEE Transactions on Industrial Informatics, vol. 15, no. 4, pp. 2405–2415, 2018.
[54] C. Verdouw, B. Tekinerdogan, A. Beulens, and S. Wolfert, "Digital twins in smart farming," Agricultural Systems, vol. 189, p. 103046, 2021.
[55] J. B. Hauge, M. Zafarzadeh, Y. Jeong, Y. Li, W. A. Khilji, C. Larsen, and M. Wiktorsson, "Digital twin testbed and practical applications in production logistics with real-time location data," International Journal of Industrial Engineering and Management, vol. 12, no. 2, p. 129, 2021.
[56] M. Stan, T. Borangiu, and S. Răileanu, "Data-and model-driven digital twins for design and logistics control of product distribution," in 2021 23rd International Conference on Control Systems and Computer Science (CSCS). IEEE, 2021, pp. 33–40.
[57] R. G. Sargent, "Model verification and validation," in Modeling and simulation in the systems engineering life cycle. Springer, 2015, pp. 57–65.

B

Python programming code

B.1. Physical Twin

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jun 23 16:00:33 2022

@author: klm92970 - Jorrit Sijtsma - 5349664
"""
"""--- Import Packages ---"""
from datetime import datetime, timedelta
import salabim as sim
import numpy as np
import scipy.stats as st
import pandas as pd
from tqdm import tqdm
import matplotlib.pyplot as plt
import warnings
from Digital_Twin import DynamicBusinessRuling


"""--- Simulation settings ---"""
EnableDigitalTwin = False
Animate = False
ExtractExposures = True                                                         #Extracts
    exposure of each ULD in a .txt; does affect performance significantly
input_dataset = "C://Users//klm92970/OneDrive - Air France KLM/Data/INPUT_DATA_SPRING_SUMMER.
    txt"
tempinput_dataset = "C://Users//klm92970/OneDrive - Air France KLM/Data/TEMPINPUT_DATA.txt"
n = 20
          #Number of simulation runs

RandomSeed = True
Suppress_trace_linenumbers = True
warnings.filterwarnings(action='ignore', category=FutureWarning)
           #Remove FutureWarning about append to reduce output clutter
"""--- Simulation Parameters ---"""
CommercialTTime = timedelta(hours=8)

MinConnectionTimeMTDToTransport = 270 #minutes
MinConnectionTimeMTD = 300 #minutes
LandSideTime = 60 #minutes
ProcessTimeMTDPCHSIn = 30 #minutes

PCHSRemoval = 300   #minutes
KC01Removal = 180  #minutes

NrTransportersKC01 = 1
NrTransportersPCHS = 2
NrTractors = 5

```

```python

COLsetpoint = 5
CRTsetpoint = 20

COLCapacity = 6
CRTCapacity = 28

TPTList = []


MissedFlight = []
WasCoolStored = []
WasSwapped = []

"""--- Useful Functions ---"""
if RandomSeed:                                                      #Assign a random seed if switch is set to True
    randomseed = '*'
else:
    randomseed = None

dateparse = lambda x: datetime.strptime(x, '%Y-%m-%d %H:%M:%S')

def detNewUTBLTB(newProdCode: str):
    if newProdCode == 'S53' or newProdCode == 'C53':
        newUTB = 25
        newLTB = 15
    elif newProdCode == 'S51' or newProdCode == 'C51':
        newUTB = 8
        newLTB = 2
    else:
        newUTB = 25
        newLTB = 2
    return newUTB, newLTB

def detExposure(ExposureTime, ExposureTemp, UTB, LTB, ULDname):
    Exposure = pd.DataFrame(
        {'Timestamp': ExposureTime,
         'Temperature': ExposureTemp,
         'UTB': UTB,
         'LTB': LTB,
         'TimeDelta': timedelta(hours = 0),
         'AMT': 0,
         'ExposureDegHr': 0,
         'ULD': ULDname})

    Exposure['Timestamp'] = Exposure['Timestamp'].dt.round("S")

    for i in range(len(Exposure.index)-1):
        if Exposure['Temperature'].loc[i] > UTB or Exposure['Temperature'].loc[i] < LTB:
           Exposure.iloc[i+1,4] = Exposure.iloc[i+1,0] - Exposure.iloc[i,0]
           if Exposure['Temperature'].loc[i] > UTB:
               Exposure.iloc[i+1,5] = (abs(Exposure.iloc[i,1] - Exposure.iloc[i,2]) + abs(Exposure.iloc[i+1,1] - Exposure.iloc[i+1,2]))/2
               Exposure.iloc[i+1,6] = Exposure.iloc[i+1,5] * float(((Exposure.iloc[i+1,4]).total_seconds()/3600))
           elif Exposure['Temperature'].loc[i] < LTB:
               Exposure.iloc[i+1,5] = (abs(Exposure.iloc[i,1] - Exposure.iloc[i,3]) + abs(Exposure.iloc[i+1,1] - Exposure.iloc[i+1,3]))/2
               Exposure.iloc[i+1,6] = Exposure.iloc[i+1,5] * float(((Exposure.iloc[i+1,4]).total_seconds()/3600))
    TotalExposure = Exposure['TimeDelta'].sum(axis=0)

    if ExtractExposures == True:
        directory = "C://Users//klm92970/OneDrive - Air France KLM/Python/Exposures/Summer/"
        filename = str(ULDname) + '_' + str(UTB) + ".txt"
        filepath = directory + filename
        Exposure.to_csv(filepath)
    return TotalExposure

def detHandlingDeadline1(STD, PCHS_In, ConnectionTime, LandSideTime):
```

```
    if (STD - (env.t_to_datetime(PCHS_In) - timedelta(minutes = 20))) > timedelta(minutes = (
    ConnectionTime + LandSideTime)):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detHandlingDeadline2(STD, PCHS_In, ConnectionTime, LandSideTime, ProcessTime):
    if (STD - env.t_to_datetime(PCHS_In)) > timedelta(minutes = (ConnectionTime +
    LandSideTime + ProcessTime)):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detHandlingDeadline3(STD, CurrentTime):
    if STD - CurrentTime > timedelta(minutes = 80):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detCCDeadline1(PCHS_In, ATA):
    if (env.t_to_datetime(PCHS_In) - ATA) < timedelta(hours = 2):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detCCDeadline2(Cool_In, ATA):
    if (env.t_to_datetime(Cool_In) - ATA) < timedelta(hours = 3):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detCCDeadline3(STD, Cool_Out):
    if  STD - env.t_to_datetime(Cool_Out) <= timedelta(minutes = KC01Removal):
        Deadline = 'OK'
    else:
        Deadline = 'NOT OK'
    return Deadline

def detAirsideLaneTime(DepartureTime):
    AirsideDistributionTime = Airsidelane_waiting_distr.sample()
    WaitingTime = AirsideDistributionTime
    return abs(WaitingTime)

def detRRTime(DepartureTime):
    RRDistributionTime = RR_distr.sample()
    RRTime = RRDistributionTime
    return abs(RRTime)

def detPCHSRemoval(DepartureTime,PCHSRemoval):
    if env.t_to_datetime(env.now()) < DepartureTime - timedelta(minutes = PCHSRemoval):
        PCHSdistr = PCHSout_distr.sample()

        while env.t_to_datetime(env.now()) >= DepartureTime - (timedelta(minutes =
    PCHSRemoval) - (timedelta(minutes = PCHSdistr))) or env.t_to_datetime(env.now()) >=
    DepartureTime - (timedelta(minutes = PCHSRemoval) + abs(timedelta(minutes = PCHSdistr))):
            PCHSdistr = PCHSout_distr.sample()

        if PCHSdistr > 0:
            PCHSRemovalTime = env.datetime_to_t((DepartureTime - timedelta(minutes =
    PCHSRemoval) - timedelta(minutes = PCHSdistr)))
        else:
            PCHSRemovalTime = env.datetime_to_t((DepartureTime - timedelta(minutes =
    PCHSRemoval) + abs(timedelta(minutes = PCHSdistr))))
    else:
        PCHSRemovalTime = env.now()
```

```
    return PCHSRemovalTime

def detKC01Removal(DepartureTime,KC01Removal):
    if env.t_to_datetime(env.now()) < DepartureTime - timedelta(minutes = KC01Removal):
        KC01distr = KC01out_distr.sample()

        while env.t_to_datetime(env.now()) >= DepartureTime - (timedelta(minutes =
    KC01Removal) - (timedelta(minutes = KC01distr))) or env.t_to_datetime(env.now()) >=
    DepartureTime - (timedelta(minutes = KC01Removal) + abs(timedelta(minutes = KC01distr))):
            KC01distr = KC01out_distr.sample()

        if KC01distr > 0:
            KC01RemovalTime = env.datetime_to_t((DepartureTime - timedelta(minutes =
    KC01Removal) - timedelta(minutes = KC01distr)))
        else:
            KC01RemovalTime = env.datetime_to_t((DepartureTime - timedelta(minutes =
    KC01Removal) + abs(timedelta(minutes = KC01distr))))
    else:
        KC01RemovalTime = env.now()

    return KC01RemovalTime


def detFlightOutSchedule(data):
    """
    Function to generate schedule of departing flights
    with corresponding freight units assigned to these
    flights + departure dates.
    """
    input_df = pd.read_csv(data, sep=",",parse_dates=['Sched_Out', 'Actual_In'], date_parser=
    dateparse)
    FlightOutSchedule = input_df[['Flight_Out','Sched_Out', 'ULD_In']]
    FlightOutSchedule = FlightOutSchedule.groupby(['Flight_Out','Sched_Out'])['ULD_In'].
    unique().reset_index()
    FlightOutSchedule = FlightOutSchedule.sort_values(by='Sched_Out', ascending=True)
    FlightOutSchedule.reset_index(inplace = True, drop = True)
    FlightOutSchedule.insert(0, 'Group_ID', range(0, len(FlightOutSchedule)))
    return FlightOutSchedule


def detNextDepGroups(schedule):
    CurrentDay = env.t_to_datetime(env.now()).date()
    NextRows = schedule[schedule['Sched_Out'].dt.date == CurrentDay]
    NextGroups = NextRows['Group_ID']
    return NextGroups

def detAvgTimedelta(data):
    total_duration = timedelta(0)
    for td in data:
        total_duration += td
    AverageExposure = (total_duration / len(data))
    return AverageExposure

def detCurrentlyStored(COL,CRT):
    COLstored = []
    CRTstored = []
    for uld in COL:
        COLstored.append(uld)
    for uld in CRT:
        CRTstored.append(uld)

    CurrentlyStoredCOL = pd.DataFrame(
        {'ULD': COLstored,
         'SHC': '',
         'Exposure': timedelta(minutes = 0),
         'STD': ''})
    CurrentlyStoredCRT = pd.DataFrame(
        {'ULD': CRTstored,
         'SHC': '',
         'Exposure': timedelta(minutes = 0),
         'STD': ''})
```

```

    for uld in CurrentlyStoredCOL['ULD']:
        index_label = CurrentlyStoredCOL.index[CurrentlyStoredCOL['ULD'] == uld]
        # if pd.isna(uld) == False:
        CurrentlyStoredCOL.loc[index_label, 'SHC'] = uld.SHC
        CurExposure = detExposure(uld.ExposureTime, uld.ExposureTemp, uld.UTB, uld.LTB, uld.name())
        CurrentlyStoredCOL.loc[index_label, 'Exposure'] = CurExposure
        CurrentlyStoredCOL.loc[index_label, 'STD'] = uld.STD
    CurrentlyStoredCOL['STD'] = pd.to_datetime(CurrentlyStoredCOL['STD'])

    for uld in CurrentlyStoredCRT['ULD']:
        index_label = CurrentlyStoredCRT.index[CurrentlyStoredCRT['ULD'] == uld]
        # if pd.isna(uld) == False:
        CurrentlyStoredCRT.loc[index_label, 'SHC'] = uld.SHC
        CurExposure = detExposure(uld.ExposureTime, uld.ExposureTemp, uld.UTB, uld.LTB, uld.name())
        CurrentlyStoredCRT.loc[index_label, 'Exposure'] = CurExposure
        CurrentlyStoredCRT.loc[index_label, 'STD'] = uld.STD
    CurrentlyStoredCRT['STD'] = pd.to_datetime(CurrentlyStoredCRT['STD'])

    return CurrentlyStoredCOL, CurrentlyStoredCRT


"""--- CLASS Definitions ---"""
#Generating model input according to input data file
class InputGenerator (sim.Component):                                     #ULDs are generated according to ATA at the hub.
    def process(self):
        df = pd.read_csv(input_dataset, sep=",",parse_dates=['Sched_Out', 'Actual_In'], date_parser=dateparse)
        AWBList = []
        for index, row in df.iterrows():
            newATA = row['Actual_In']
            newSTD = row['Sched_Out']
            newSHC = row['SHC']
            newProdCode = row['Product_Code']
            newUTB, newLTB = detNewUTBLTB(newProdCode)
            newULDCode = row['ULD_In']
            newFlightIn = row['Flight_In']
            newFlightOut = row['Flight_Out']
            AWBstring = row['AWBs']
            newAWB = np.fromstring(AWBstring[1:-1], sep=' ')

            #Hold until a new shipment is created according to Actual_In in the input data
            yield self.hold(till=env.datetime_to_t(newATA))
            newFreightUnit = FreightUnit(env=self.env, SHC = newSHC, ProdCode = newProdCode, UTB = newUTB, LTB = newLTB, ATA = newATA, TransitTime = None, STD = newSTD,ULDCode = newULDCode,
                                         Flight_Out = newFlightOut, ExposureTime = [], ExposureTemp = [], Exposure = None, TPT = None, TIR = None, Status = None,
                                         CoolStorage = None, Actual_Out = None, PCHS_In = None, Cool_In = None, Cool_Out = None, CCDeadline1 = None, CCDeadline2 = None,
                                         CCDeadline3 = None, HDeadline1 = None, HDeadline2 = None, HDeadline3 = None)
            newFreightUnit.name(newULDCode)
            newFreightUnit.enter(truckarrival.EntranceQueue)
            newFreightUnit.enter(AllFreightUnits)
            newFreightUnit.enter(FreightUnitsInProcess)

            #For each line, extract the indicated AWBs, generate an instance and add the corresponding ULD number to it.
            #In the case of multiple AWBs on 1 ULD, the same ULD code is added to each AWB.
            for i in range(len(newAWB)):
                newAWBNumber = str(int(newAWB[i]))

                if newAWBNumber not in AWBList:
                    newAirWayBill = AirWayBill(env=self.env,AWBNumber = newAWBNumber, ULDCode = [], myULDs = [])
                    newAirWayBill.name(newAWBNumber)
                    newAirWayBill.ULDCode.append(newULDCode)
```

```
                    AWBList.append(newAirWayBill.name())
                    newAirWayBill.enter(AllAWBs)
                #In the case of multiple ULDs for 1 AWB, a list of corresponding ULD codes is generated.
                elif newAWBNumber in AWBList:
                    for i in AllAWBs:
                        if i.name() == newAWBNumber:
                            AWBtochange = i
                    AWBtochange.ULDCode.append(newULDCode)
                    AWBtochange.activate()

            #Extract the next Flight In from the dataset
            if index + 1 < df.shape[0]:  # Check if index+1 does not exceed dataframe size
                next_line = df.iloc[[index + 1]]
                next_line = next_line.squeeze(axis=0)
                nextFlightIn = next_line['Flight_In']
                nextATA = next_line['Actual_In']
            #Activate truckarrival for the last incoming ULD
            elif index + 1 == df.shape[0]:
                truckarrival.activate()

            # Check whether the Flightin is not equal to the next line
            if (((nextFlightIn != newFlightIn) and (nextATA != newATA)) or ((nextFlightIn == newFlightIn) and (nextATA != newATA))):
                truckarrival.activate()

class TempGenerator(sim.Component):
    def setup(self, AmbientTemperature = None, PCHSTemperature = None, PCHSInTemperature = None, PCHSOutTemperature = None):
        self.AmbientTemperature = AmbientTemperature
        self.PCHSTemperature = PCHSTemperature
        self.PCHSInTemperature = PCHSInTemperature
        self.PCHSOutTemperature = PCHSOutTemperature
    def process(self):
        temp_df = pd.read_csv(tempinput_dataset, sep=",",parse_dates=['datetime'], date_parser=dateparse)
        for index, row in temp_df.iterrows():
            newDateTime = row['datetime']

            yield self.hold(till = env.datetime_to_t(newDateTime))

            #Push current ambient temperature to all relevant ULDs (not in PCHS or KC01)
            self.AmbientTemperature = row['T']
            env.print_trace('***INFO***','Amb temp update:',str(self.AmbientTemperature))
            env.print_trace('***INFO***','PCHS temp update:',str(self.PCHSTemperature))
            self.PCHSInTemperature = row['[71]']
            self.PCHSOutTemperature = row['[22]']
            self.PCHSTemperature = row['[24]']



            for uld in FreightUnitsInProcess:
                if uld in AmbientConditions:
                    uld.ExposureTemp.append(self.AmbientTemperature)
                    uld.ExposureTime.append(newDateTime)
                elif uld in storage_PCHS:
                    uld.ExposureTemp.append(self.PCHSTemperature)
                    uld.ExposureTime.append(newDateTime)


class AirWayBill(sim.Component):
    def setup(self, AWBNumber, ULDCode, myULDs, myTOR = [], myExposure = [], myMissedFlights = []):
        self.AWBNumber = AWBNumber
        self.ULDCode = ULDCode
        self.myULDs = myULDs
        self.myTOR = myTOR
        self.myExposure = myExposure
        self.myMissedFlights = myMissedFlights
    def process(self):
        while True:
```

```
            #Extract ULD instances to connect to AWB
            for i in self.ULDCode:
                for j in AllFreightUnits:
                    if i == j.name() and j not in self.myULDs:
                        self.myULDs.append(j)
            yield self.passivate()

class FreightUnit(sim.Component):
    def setup(self, SHC, ProdCode, UTB, LTB, ATA, TransitTime, STD, ULDCode, Flight_Out, ExposureTime,
              ExposureTemp, Exposure, TPT, TIR, Status, CoolStorage, Actual_Out, PCHS_In, Cool_In, Cool_Out, CCDeadline1,
              CCDeadline2, CCDeadline3, HDeadline1, HDeadline2, HDeadline3):
        self.SHC = SHC
        self.ProdCode = ProdCode
        self.UTB = UTB
        self.LTB = LTB
        self.TransitTime = TransitTime
        self.STD = STD
        self.ULDCode = ULDCode
        self.Flight_Out = Flight_Out
        self.Status = Status
        self.CoolStorage = CoolStorage
        self.ATA = ATA
        self.Actual_Out = Actual_Out
        self.PCHS_In = PCHS_In
        self.Cool_In = Cool_In
        self.Cool_Out = Cool_Out
        self.ExposureTime = ExposureTime
        self.ExposureTemp = ExposureTemp
        self.Exposure = Exposure
        self.TPT = TPT
        self.TIR = TIR
        self.CCDeadline1 = CCDeadline1
        self.CCDeadline2 = CCDeadline2
        self.CCDeadline3 = CCDeadline3
        self.HDeadline1 = HDeadline1
        self.HDeadline2 = HDeadline2
        self.HDeadline3 = HDeadline3
        #for each freight unit, record the TOR, total processing time,
        #whether processing deadlines are met (both cool chain and handovers), and whether exposure is OK or NOT OK

        #Total processing time (TPT) = Actual Out - Actual In
        #Exposure = Total time outside of SHC temp range

        #Cool Chain - OK / NOT OK:
        #1* ULDs into PCHS within 2 hrs after ARR --> PCHS_In & Actual_In
        #2* ULDs (lead time > 8hrs) to KC01 within 3 hrs after ARR --> Cool_In & Actual_In & CoolStorage
        #3* ULDs (lead time > 8hrs) removed from KC01 no longer than 3 hrs before STD --> Cool_Out & STD & CoolStorage

        #Handling deadlines:
        #1* Shipments received on time into MTD process (Transit flow: min connection time + 60 mins Landside time, min connection time: EUR - ICA = 300 mins before departure baseline)
        #2* Shipments on time handed over from MTD to Transport (min connection times + 60 min landside time + 30 min (process time MTD --> PCHS IN), min connection time: EUR - ICA = 270 mins before departure baseline)
        #3* Shipments on time handed over from Transport to GS (80 mins before baseline departure)

    def animation_objects(self, id):
        '''
        the way the component is determined by the id, specified in AnimateQueue
        'text' means just the name
        any other value represents the colour
        '''
        if id == 'text':
            ao0 = sim.AnimateText(text=self.name(), textcolor='fg', text_anchor='nw')
```

```
            return 0, 16, ao0
        else:
            ao0 = sim.AnimateRectangle((-20, 0, 70, 20),
                text=self.name(), fillcolor=id, textcolor='white', arg=self)
            return 95, 0, ao0

    def process(self):
        if self.STD - env.t_to_datetime(env.now()) < timedelta(hours = 6):     #min connection time + 60 min landside time = 360 min / 6 hours
            self.CoolStorage = 'DirectToAirside'

        ###### Passivate until activation by TruckArrival ######
        yield self.passivate()
        self.TransitTime = self.STD - env.t_to_datetime(env.now())

        ###### Verify if not late arrival ######
        if self.CoolStorage != 'DirectToAirside':
            self.enter(decisionmodule.DecisionMakingQueue)
            if decisionmodule.ispassive():
                decisionmodule.activate()
            yield self.passivate()
        env.print_trace('**INFO**',FreightUnit.name(self),'CoolStorage:',self.CoolStorage,self.SHC)

        if self.CoolStorage == 'DirectToAirside':
            env.print_trace('**INFO**',FreightUnit.name(self),'Late Arrival:',self.CoolStorage)

            ###### Calculate deadlines ######
            self.HDeadline1 = detHandlingDeadline1(self.STD, (env.datetime_to_t(env.t_to_datetime(env.now()) + timedelta(minutes = 20))), MinConnectionTimeMTD, LandSideTime)  #Handling Deadline 1


            self.HDeadline2 = detHandlingDeadline2(self.STD, env.now(), MinConnectionTimeMTDToTransport, LandSideTime, ProcessTimeMTDPCHSIn)  #Handling Deadline 2


            self.ExposureTemp.append(tempgenerator.PCHSInTemperature)
            self.ExposureTime.append(env.t_to_datetime(env.now()))

            ###### Request tractor ######
            while len(AvailableTractorsQueue) == 0:
                yield self.standby()
            myTractor = AvailableTractorsQueue.pop()
            myTractor.FUtoRR = self
            myTractor.activate()
            yield self.passivate()

            ###### Buffer at air side lane ######
            self.enter(AirsideLaneQueue)
            WaitingTime = detAirsideLaneTime(self.STD)
            yield self.hold(WaitingTime)
            self.leave(AirsideLaneQueue)

            ###### Request tractor for ramp ride ######
            self.CoolStorage = ''                                              # Reset status for correct tractor ride
            while len(AvailableTractorsQueue) == 0:
                yield self.standby()
            myTractor = AvailableTractorsQueue.pop()
            myTractor.FUtoRR = self
            myTractor.activate()
            yield self.passivate()
            self.CoolStorage = 'DirectToAirside'

            self.HDeadline3 = detHandlingDeadline3(self.STD,env.t_to_datetime(env.now()))  #Handling Deadline 3


```

```
            ###### Waiting and loading until departure ######
            if env.t_to_datetime(env.now()) > self.STD:
                MissedFlight.append(self)
            else:
                yield self.hold(till = env.datetime_to_t(self.STD))

            self.ExposureTemp.append(tempgenerator.AmbientTemperature)
            self.ExposureTime.append(env.t_to_datetime(env.now()))

            self.leave(AmbientConditions)
            self.leave(FreightUnitsInProcess)
            self.Actual_Out = env.now()

            self.TPT = env.t_to_datetime(self.Actual_Out) - self.ATA
            TPTList.append((self.TPT).total_seconds()/3600)
            TPTMonitor.tally((self.TPT).total_seconds()/3600)
            env.print_trace('**TPT**',FreightUnit.name(self),str(self.TPT))

            self.TIR = timedelta(minutes=0) # No Cool Chain resources --> no cool storage.
            env.print_trace('**TIR**',FreightUnit.name(self),str(self.TIR))

            self.enter(HandledFreightUnits)

            self.Exposure = detExposure(self.ExposureTime,self.ExposureTemp,self.UTB,self.LTB
,self.name())
            env.print_trace('**Exposure*',FreightUnit.name(self),str(self.Exposure))


        else:
            ###### Enter PCHS ######
            self.enter(pchsentrance.pchs_in_queue)
            if pchsentrance.ispassive():
                pchsentrance.activate()
            yield self.passivate()

            ###### Calculate deadlines ######
            self.HDeadline1 = detHandlingDeadline1(self.STD, self.PCHS_In,
MinConnectionTimeMTD, LandSideTime)  #Handling Deadline 1

            self.HDeadline2 = detHandlingDeadline2(self.STD, self.PCHS_In,
MinConnectionTimeMTDToTransport, LandSideTime, ProcessTimeMTDPCHSIn)  #Handling Deadline
2

            self.CCDeadline1 = detCCDeadline1(self.PCHS_In, self.ATA)   #Cool Chain Deadline
1

            ##### Stored in PCHS until removal ######
            self.enter(controller.ControllerQueue)
            if controller.ispassive():
                controller.activate()
            yield self.passivate()



            ###### Remove from PCHS storage ######
            self.enter(pchsexit.PCHSExitQueue)
            if pchsexit.ispassive():
                pchsexit.activate()
            yield self.passivate()

            if self.CoolStorage == 'Yes':
                ###### Request transporter to KC01 ######
                self.Status = 'In'
                while len(AvailableTransportersQueueKC01) == 0:
                    yield self.standby()
                myTransporter = AvailableTransportersQueueKC01.pop()
                myTransporter.FUtotransport = self
                myTransporter.activate()
                yield self.passivate()

                ###### Enter KC01 ######
```

```python
            self.enter(kc01.KC01ToDo)
            if kc01.ispassive():
                kc01.activate()
            yield self.passivate()

            ###### Calculate deadline ######
            self.CCDeadline2 = detCCDeadline2(self.Cool_In, self.ATA)


            ###### Stored in KC01 until removal ######
            self.enter(controller.ControllerQueue)
            if controller.ispassive():
                controller.activate()
            yield self.passivate()

            ###### Remove from KC01 storage ######
            self.Status = 'Out'
            kc01.KC01ToDo.add_sorted(self,1)                               # Priority for outgoing ULDs KC01
            if kc01.ispassive():
                kc01.activate()
            yield self.passivate()

            ###### Calculate deadline ######
            self.CCDeadline3 = detCCDeadline3(self.STD, self.Cool_Out)
            env.print_trace(self.name(),str(self.STD - env.t_to_datetime(self.Cool_Out)))

            ###### Request transporter to air side lane ######
            while len(AvailableTransportersQueueKC01) == 0:
                yield self.standby()
            myTransporter = AvailableTransportersQueueKC01.pop()
            myTransporter.FUtotransport = self
            myTransporter.activate()
            yield self.passivate()

        elif self.CoolStorage == 'No':
            ###### Request transporter to air side lane ######
            while len(AvailableTransportersQueuePCHS) == 0:
                yield self.standby()
            self.Status = 'PCHS'
            myTransporter = AvailableTransportersQueuePCHS.pop()
            myTransporter.FUtotransport = self
            myTransporter.activate()
            yield self.passivate()

        ###### Buffer at air side lane ######
        self.enter(AirsideLaneQueue)
        WaitingTime = detAirsideLaneTime(self.STD)
        yield self.hold(WaitingTime)
        self.leave(AirsideLaneQueue)

        ###### Request tractor for ramp ride ######
        while len(AvailableTractorsQueue) == 0:
            yield self.standby()
        myTractor = AvailableTractorsQueue.pop()
        myTractor.FUtoRR = self
        myTractor.activate()
        yield self.passivate()

        ###### Calculate deadline ######
        self.HDeadline3 = detHandlingDeadline3(self.STD,env.t_to_datetime(env.now())) #Handling Deadline 3

        if env.t_to_datetime(env.now()) > self.STD:
            env.print_trace('!!!ALERT!!!',str(self.name()),'MISSED FLIGHT')
            MissedFlight.append(self)
        else:
            yield self.hold(till = env.datetime_to_t(self.STD))

        self.ExposureTemp.append(tempgenerator.AmbientTemperature)
        self.ExposureTime.append(env.t_to_datetime(env.now()))
```

```python


            self.leave(AmbientConditions)
            self.leave(FreightUnitsInProcess)
            self.Actual_Out = env.now()

            self.TPT = env.t_to_datetime(self.Actual_Out) - self.ATA
            TPTList.append((self.TPT).total_seconds()/3600)
            TPTMonitor.tally((self.TPT).total_seconds()/3600)
            env.print_trace('**TPT**',FreightUnit.name(self),str(self.TPT))

            if self.CoolStorage == 'No':
                self.TIR = timedelta(minutes=0)
            else:
                self.TIR = env.t_to_datetime(self.Cool_Out) - env.t_to_datetime(self.Cool_In)
            env.print_trace('**TIR**',FreightUnit.name(self),str(self.TIR))

            self.enter(HandledFreightUnits)

            self.Exposure = detExposure(self.ExposureTime,self.ExposureTemp,self.UTB,self.LTB
    ,self.name())
            env.print_trace('**Exposure*',FreightUnit.name(self),str(self.Exposure))


class Controller(sim.Component):
    def setup(self, FlightOutSchedule = detFlightOutSchedule(input_dataset), myULDs = [],
    myDepGroups = []):
        self.FlightOutSchedule = FlightOutSchedule
        self.myULDs = myULDs
        self.myDepGroups = myDepGroups

        self.ActiveDEPGroups = sim.Queue('Currently active DEP Groups')
        self.ControllerQueue = sim.Queue('ULDs requesting removal time')
    def process(self):
        while True:
            while len(self.ControllerQueue) == 0:
                yield self.passivate()

            ULDToAssign = self.ControllerQueue.pop()

            ###### Request: PCHS removal time ######
            if ULDToAssign in storage_PCHS:

                if ULDToAssign.CoolStorage == 'No':
                    RemovalTime = detPCHSRemoval(ULDToAssign.STD, PCHSRemoval)
                    ULDToAssign.hold(till = RemovalTime)

                elif ULDToAssign.CoolStorage == 'Yes':
                        RemovalTime = PCHSin_KC01in_distr.sample()
                        ULDToAssign.hold(till = env.datetime_to_t(env.t_to_datetime(env.now()
    ) + timedelta (minutes = RemovalTime)))

            ###### Request: KC01 removal time ######
            elif ULDToAssign in storage_KC01_COL:
                if ULDToAssign.CoolStorage == 'PharmaSwap' and ULDToAssign.Status != 'Out':
                    ULDToAssign.activate() #Remove now
                elif ULDToAssign.CoolStorage == 'PharmaSwap' and ULDToAssign.Status == 'Out':
                    continue
                else:
                    RemovalTime = detKC01Removal(ULDToAssign.STD, KC01Removal)
                    ULDToAssign.hold(till = RemovalTime)

            elif ULDToAssign in storage_KC01_CRT:
                if ULDToAssign.CoolStorage == 'PharmaSwap' and ULDToAssign.Status != 'Out':
                    ULDToAssign.activate() #Remove now
                elif ULDToAssign.CoolStorage == 'PharmaSwap' and ULDToAssign.Status == 'Out':
                    continue
                else:
                    RemovalTime = detKC01Removal(ULDToAssign.STD, KC01Removal)
                    ULDToAssign.hold(till = RemovalTime)

```

```python
            elif ULDToAssign.CoolStorage == 'Out':
                continue


class TruckArrival(sim.Component):
    def setup(self, myULDs = []):
        self.EntranceQueue = sim.Queue('Shipment Arrival Queue')
        self.myULDs = myULDs
    def process(self):
        while True:
            while len(self.EntranceQueue) == 0:
                yield self.passivate()

            LateArrivals = []
            #select all ULDs arriving on the same truck
            for i in self.EntranceQueue:
                ULDToAppend = i
                LateArrivals.append(i.CoolStorage)
                self.myULDs.append(ULDToAppend)
                self.EntranceQueue.remove(i)

            if 'DirectToAirside' in LateArrivals:
                WaitingTime = 15
            else:
                #Generate waiting time according to distribution for all selected ULDs
                WaitingTime = Arrival_distr.sample()

            #Assign waiting + unloading time to ULDs
            for i in self.myULDs:
                i.hold(WaitingTime)

            self.myULDs.clear()
            LateArrivals.clear()

class DecisionModule(sim.Component):                                           # Transittime, SHC & weather alarm in the base / current situation
    def setup(self, COLPrevElement = [], CRTPrevElement = []):
        self.DecisionMakingQueue = sim.Queue('Apply Business Ruling queue')
        self.COLAssigned = sim.Queue('ULDs assigned to COL storage')
        self.CRTAssigned = sim.Queue('ULDs assigned to CRT storage')
        self.COLPrevElement = COLPrevElement
        self.CRTPrevElement = CRTPrevElement
    def process(self):
        if EnableDigitalTwin == False:
            while True:
                while len(self.DecisionMakingQueue) == 0:
                    yield self.passivate()


                #Verify whether storage is full each time upon activation
                if len(storage_KC01_COL) + len(self.COLAssigned) >= COLCapacity:
                    COLStorageFull.set(value = True)
                    env.print_trace('**!!!**','COL storage full!')
                    env.print_trace(str(len(storage_KC01_COL)))


                if len(storage_KC01_CRT) + len(self.CRTAssigned) >= CRTCapacity:
                    CRTStorageFull.set(value = True)
                    env.print_trace('**!!!**','CRT storage full!')
                    env.print_trace(str(len(storage_KC01_CRT)))

                new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((COLCapacity - len(storage_KC01_COL)) / COLCapacity),'#Stored': len(storage_KC01_COL)}]
                kc01.COLAvailability = kc01.COLAvailability.append(new_entry, ignore_index = True)

                new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((CRTCapacity - len(storage_KC01_CRT)) / CRTCapacity), '#Stored': len(storage_KC01_CRT)}]
                kc01.CRTAvailability = kc01.CRTAvailability.append(new_entry, ignore_index = True)
```

```

            # select freightunit for decision making
            myFreightUnit = self.DecisionMakingQueue.pop()

            # Assign CoolStorage status based on business ruling
            if myFreightUnit.SHC == 'COL':
                if myFreightUnit.TransitTime > CommercialTTime:
                    if COLStorageFull.get() == False:
                        myFreightUnit.CoolStorage = 'Yes'
                        self.COLAssigned.append(myFreightUnit)
                    #If KC01 is full, swap with a ULD which departs sooner
                    #Check if storage is actually full (instead of only assigned ULDs):
                    elif COLStorageFull.get() == True and len(storage_KC01_COL) != 0:
                        env.print_trace('**!!!**','Unable to store',str(myFreightUnit), myFreightUnit.SHC)
                        #Find the ULD in storage with earliest STD
                        min_ele = storage_KC01_COL[0]
                        for i in storage_KC01_COL:
                            if i in self.COLPrevElement:
                                if i == min_ele:
                                    min_ele = storage_KC01_COL.successor(i)
                                continue

                            if i.STD < min_ele.STD:
                                min_ele = i
                        #Check whether there is actually a ULD to swap with
                        if min_ele == None or myFreightUnit.STD <= min_ele.STD:
                            myFreightUnit.CoolStorage = 'No'
                        #In the case that there is a ULD to swap
                        elif myFreightUnit.STD > min_ele.STD:
                            myFreightUnit.CoolStorage = 'Yes'
                            self.COLAssigned.append(myFreightUnit)
                            env.print_trace('**!!!**','','removing to make space',min_ele.name())
                            min_ele.CoolStorage = 'PharmaSwap'
                            min_ele.enter(controller.ControllerQueue)
                            WasSwapped.append(min_ele)
                            if controller.ispassive():
                                controller.activate()
                            #Add the ULD to swap to the list of ULDs which are chosen to swap
                            if min_ele not in self.COLPrevElement:
                                self.COLPrevElement.append(min_ele)
                    #No swapping of ULDs in the case that storage is full but only because the maximum of ULDs to KC01 has already been assigned
                    else:
                        myFreightUnit.CoolStorage = 'No'
                else:
                    myFreightUnit.CoolStorage = 'No'


            elif myFreightUnit.SHC == 'CRT':
                if myFreightUnit.TransitTime > CommercialTTime:
                    if CRTStorageFull.get() == False:
                        myFreightUnit.CoolStorage = 'Yes'
                        self.CRTAssigned.append(myFreightUnit)
                    elif CRTStorageFull.get() == True and len(storage_KC01_CRT) != 0:
                        env.print_trace('**!!!**','Unable to store',str(myFreightUnit), myFreightUnit.SHC)

                        min_ele = storage_KC01_CRT[0]
                        for i in storage_KC01_CRT:
                            if i in self.CRTPrevElement:
                                if i == min_ele:
                                    min_ele = storage_KC01_CRT.successor(i)
                                continue

                            if i.STD < min_ele.STD:
                                min_ele = i

                        if min_ele == None or myFreightUnit.STD <= min_ele.STD:
```

```
                            myFreightUnit.CoolStorage = 'No'

                        elif myFreightUnit.STD > min_ele.STD:
                            myFreightUnit.CoolStorage = 'Yes'
                            self.CRTAssigned.append(myFreightUnit)
                            env.print_trace('**!!!**','','removing to make space',min_ele.name())
                            min_ele.CoolStorage = 'PharmaSwap'
                            min_ele.enter(controller.ControllerQueue)
                            WasSwapped.append(min_ele)
                            if controller.ispassive():
                                controller.activate()
                            if min_ele not in self.CRTPrevElement:
                                self.CRTPrevElement.append(min_ele)
                    else:
                        myFreightUnit.CoolStorage = 'No'
                else:
                    myFreightUnit.CoolStorage = 'No'

            elif myFreightUnit.SHC == 'ERT':
                if tempgenerator.AmbientTemperature < 5 or tempgenerator.AmbientTemperature >= 18:          #In case of weather alarm: ERT is stored in KC01 CRT!
                    if myFreightUnit.TransitTime > CommercialTTime:
                        if CRTStorageFull.get() == False:

                            myFreightUnit.CoolStorage = 'Yes'
                            self.CRTAssigned.append(myFreightUnit)
                        elif CRTStorageFull.get() == True and len(storage_KC01_CRT) != 0:
                            env.print_trace('**!!!**','Unable to store',str(myFreightUnit),myFreightUnit.SHC)

                            min_ele = storage_KC01_CRT[0]
                            for i in storage_KC01_CRT:
                                if i in self.CRTPrevElement:
                                    if i == min_ele:
                                        min_ele = storage_KC01_CRT.successor(i)
                                    continue

                                if i.STD < min_ele.STD:
                                    min_ele = i

                            if min_ele == None or myFreightUnit.STD <= min_ele.STD:
                                myFreightUnit.CoolStorage = 'No'

                            elif myFreightUnit.STD > min_ele.STD:
                                myFreightUnit.CoolStorage = 'Yes'
                                self.CRTAssigned.append(myFreightUnit)
                                env.print_trace('**!!!**','','removing to make space',min_ele.name())
                                min_ele.CoolStorage = 'PharmaSwap'
                                min_ele.enter(controller.ControllerQueue)
                                WasSwapped.append(min_ele)
                                if controller.ispassive():
                                    controller.activate()
                                if min_ele not in self.CRTPrevElement:
                                    self.CRTPrevElement.append(min_ele)
                        else:
                            myFreightUnit.CoolStorage = 'No'
                    else:
                        myFreightUnit.CoolStorage = 'No'
                else:
                    myFreightUnit.CoolStorage = 'No'

            myFreightUnit.activate()

    if EnableDigitalTwin == True:
        while True:
            while len(self.DecisionMakingQueue) == 0:
                yield self.passivate()
            #Verify whether storage is full each time upon activation
            if len(storage_KC01_COL) + len(self.COLAssigned) >= COLCapacity:
```

```
                    COLStorageFull.set(value = True)
                    env.print_trace('**!!!**','COL storage full!')
                    env.print_trace(str(len(storage_KC01_COL)))


                if len(storage_KC01_CRT) + len(self.CRTAssigned) >= CRTCapacity:
                    CRTStorageFull.set(value = True)
                    env.print_trace('**!!!**','CRT storage full!')
                    env.print_trace(str(len(storage_KC01_CRT)))

                new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((COLCapacity - len(storage_KC01_COL)) / COLCapacity),'#Stored': len(storage_KC01_COL)}]
                kc01.COLAvailability = kc01.COLAvailability.append(new_entry, ignore_index = True)

                new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((CRTCapacity - len(storage_KC01_CRT)) / CRTCapacity), '#Stored': len(storage_KC01_CRT)}]
                kc01.CRTAvailability = kc01.CRTAvailability.append(new_entry, ignore_index = True)

                # select freightunit for decision making
                myFreightUnit = self.DecisionMakingQueue.pop()

                #Determine current exposure + time + stored ULD characteristics
                myFreightUnit.ExposureTemp.append(tempgenerator.PCHSInTemperature)
                myFreightUnit.ExposureTime.append(env.t_to_datetime(env.now()))
                CurrentULDExposure = detExposure(myFreightUnit.ExposureTime,myFreightUnit.ExposureTemp,myFreightUnit.UTB,myFreightUnit.LTB,myFreightUnit.name())
                CurrentTime = env.t_to_datetime(env.now())
                COLStored, CRTStored = detCurrentlyStored(storage_KC01_COL, storage_KC01_CRT)
                myFreightUnit.CoolStorage, ULDToRemove = DynamicBusinessRuling(CurrentTime, CurrentULDExposure, myFreightUnit.SHC, myFreightUnit.UTB, myFreightUnit.LTB,
                                                                  tempgenerator.AmbientTemperature, tempgenerator.PCHSTemperature, myFreightUnit.STD,
                                                                  COLStored, CRTStored, len(self.COLAssigned), len(self.CRTAssigned), self.COLPrevElement, self.CRTPrevElement)


                if myFreightUnit.CoolStorage == 'Yes':
                   if myFreightUnit.SHC == 'COL':
                       self.COLAssigned.append(myFreightUnit)
                   if myFreightUnit.SHC == 'CRT' or myFreightUnit.SHC == 'ERT':
                       self.CRTAssigned.append(myFreightUnit)

                if ULDToRemove != '':
                   if ULDToRemove.SHC == 'COL':
                       env.print_trace('**!!!**','','removing to make space',ULDToRemove.name())
                       ULDToRemove.CoolStorage = 'PharmaSwap'
                       ULDToRemove.enter(controller.ControllerQueue)
                       WasSwapped.append(ULDToRemove)
                       if controller.ispassive():
                           controller.activate()
                       if ULDToRemove not in self.COLPrevElement:
                           self.COLPrevElement.append(ULDToRemove)
                   elif ULDToRemove.SHC == 'CRT' or ULDToRemove.SHC == 'ERT':
                       env.print_trace('**!!!**','','removing to make space',ULDToRemove.name())
                       ULDToRemove.CoolStorage = 'PharmaSwap'
                       ULDToRemove.enter(controller.ControllerQueue)
                       WasSwapped.append(ULDToRemove)
                       if controller.ispassive():
                           controller.activate()
                       if ULDToRemove not in self.CRTPrevElement:
                           self.CRTPrevElement.append(ULDToRemove)

                myFreightUnit.activate()


class PCHSentrance(sim.Component):
    def setup(self):
```

```python
        self.pchs_in_queue = sim.Queue('PCHS Entrance Queue')
    def process(self):
        while True:
            while len(self.pchs_in_queue) == 0:
                yield self.passivate()

            FreightUnitIn = self.pchs_in_queue.pop()
            FreightUnitIn.ExposureTemp.append(tempgenerator.PCHSInTemperature)
            FreightUnitIn.ExposureTime.append(env.t_to_datetime(env.now()))
            yield self.hold(15)
            storage_PCHS.add(FreightUnitIn)
            FreightUnitIn.PCHS_In = FreightUnitIn.enter_time(storage_PCHS)
            FreightUnitIn.ExposureTemp.append(tempgenerator.PCHSTemperature) #
            FreightUnitIn.ExposureTime.append(env.t_to_datetime(env.now()))
            FreightUnitIn.activate()

class PCHSexit(sim.Component):
    def setup(self):
        self.PCHSExitQueue = sim.Queue ('PCHS exit queue')
    def process(self):
        while True:
            while len(self.PCHSExitQueue) == 0:
                yield self.passivate()

            FreightUnitOut = self.PCHSExitQueue.pop()
            FreightUnitOut.leave(storage_PCHS)
            yield self.hold(15)
            FreightUnitOut.ExposureTemp.append(tempgenerator.PCHSOutTemperature)
            FreightUnitOut.ExposureTime.append(env.t_to_datetime(env.now()))
            FreightUnitOut.enter(AmbientConditions)
            FreightUnitOut.activate()


class KC01(sim.Component):                                        # retrieval of 6 - 8 ULDs per hour max. Fresh not taken into account, so 6 is chosen.
    def setup(self, COLTemp, CRTTemp, COLAvailability = pd.DataFrame(columns = ['Timestamp','Availability','#Stored']), CRTAvailability = pd.DataFrame(columns = ['Timestamp','Availability','#Stored'])):
        self.KC01ToDo = sim.Queue('KC01 processing queue')
        self.COLTemp = COLTemp
        self.CRTTemp = CRTTemp
        self.COLAvailability = COLAvailability
        self.CRTAvailability = CRTAvailability

    def process(self):
        while True:
            while len(self.KC01ToDo) == 0:
               yield self.passivate()

            FirstFreightUnit = self.KC01ToDo.pop()
            env.print_trace('**INFO**',FirstFreightUnit.name(),FirstFreightUnit.SHC, FirstFreightUnit.Status)


            if ETVbusy.get() ==  False:                                  #check if ETV KC01 is available

                if FirstFreightUnit.Status == 'In':
                    FirstFreightUnit.leave(AmbientConditions)
                    ETVbusy.set(value = True)
                    if FirstFreightUnit.SHC == 'COL':
                        yield self.hold(10)
                        storage_KC01_COL.add(FirstFreightUnit)
                        FirstFreightUnit.leave(decisionmodule.COLAssigned)
                        FirstFreightUnit.Cool_In = FirstFreightUnit.enter_time(storage_KC01_COL)
                        FirstFreightUnit.ExposureTemp.append(self.COLTemp)
                        FirstFreightUnit.ExposureTime.append(env.t_to_datetime(FirstFreightUnit.Cool_In))
                        FirstFreightUnit.activate()

```

```
                    new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((COLCapacity - len(storage_KC01_COL)) / COLCapacity),'#Stored': len(storage_KC01_COL)}]
                    self.COLAvailability = self.COLAvailability.append(new_entry, ignore_index = True)



                elif FirstFreightUnit.SHC == 'CRT' or FirstFreightUnit.SHC == 'ERT':
                    yield self.hold(10)
                    storage_KC01_CRT.add(FirstFreightUnit)
                    FirstFreightUnit.leave(decisionmodule.CRTAssigned)
                    FirstFreightUnit.Cool_In = FirstFreightUnit.enter_time(storage_KC01_CRT)
                    FirstFreightUnit.ExposureTemp.append(self.CRTTemp)
                    FirstFreightUnit.ExposureTime.append(env.t_to_datetime(FirstFreightUnit.Cool_In))
                    FirstFreightUnit.activate()

                    new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((CRTCapacity - len(storage_KC01_CRT)) / CRTCapacity),'#Stored': len(storage_KC01_CRT)}]
                    self.CRTAvailability = self.CRTAvailability.append(new_entry, ignore_index = True)


            elif FirstFreightUnit.Status == 'Out':
                ETVbusy.set(value = True)
                if FirstFreightUnit.SHC == 'COL':
                    FirstFreightUnit.ExposureTemp.append(self.COLTemp)
                    FirstFreightUnit.ExposureTime.append(env.t_to_datetime(env.now()))
                    yield self.hold(10)
                    FirstFreightUnit.leave(storage_KC01_COL)
                    FirstFreightUnit.Cool_Out = env.now()
                    WasCoolStored.append(FirstFreightUnit)
                    FirstFreightUnit.activate()

                    new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((COLCapacity - len(storage_KC01_COL)) / COLCapacity),'#Stored': len(storage_KC01_COL)}]
                    self.COLAvailability = self.COLAvailability.append(new_entry, ignore_index = True)

                    if FirstFreightUnit in decisionmodule.COLPrevElement:
                        decisionmodule.COLPrevElement.remove(FirstFreightUnit)


                    if len(storage_KC01_COL) + len(decisionmodule.COLAssigned) >= COLCapacity:
                        COLStorageFull.set(value = True)
                        env.print_trace('**!!!**','COL storage full!')
                        env.print_trace(str(len(storage_KC01_COL)))

                    else:
                        COLStorageFull.set(value = False)


                elif FirstFreightUnit.SHC == 'CRT' or FirstFreightUnit.SHC == 'ERT':
                    FirstFreightUnit.ExposureTemp.append(self.CRTTemp)
                    FirstFreightUnit.ExposureTime.append(env.t_to_datetime(env.now()))
                    yield self.hold(10)
                    FirstFreightUnit.leave(storage_KC01_CRT)
                    FirstFreightUnit.Cool_Out = env.now()
                    WasCoolStored.append(FirstFreightUnit)
                    FirstFreightUnit.activate()

                    new_entry = [{'Timestamp': env.t_to_datetime(env.now()),'Availability': ((CRTCapacity - len(storage_KC01_CRT)) / CRTCapacity), '#Stored': len(storage_KC01_CRT)}]
                    self.CRTAvailability = self.CRTAvailability.append(new_entry, ignore_index = True)
```

```
                        if FirstFreightUnit in decisionmodule.CRTPrevElement:
                            decisionmodule.CRTPrevElement.remove(FirstFreightUnit)


                        if len(storage_KC01_CRT) + len(decisionmodule.CRTAssigned) >=
    CRTCapacity:
                            CRTStorageFull.set(value = True)
                            env.print_trace('**!!!**','CRT storage full!')
                            env.print_trace(str(len(storage_KC01_CRT)))

                        else:
                            CRTStorageFull.set(value = False)


                    FirstFreightUnit.enter(AmbientConditions)

            ETVbusy.set(value=False)


class Transporter(sim.Component):
    def setup(self, FUtotransport = None):
        self.FUtotransport = FUtotransport
    def process(self):
        while True:
            yield self.passivate()
            self.FUtotransport.ExposureTemp.append(tempgenerator.AmbientTemperature)
            self.FUtotransport.ExposureTime.append(env.t_to_datetime(env.now()))
            if self.FUtotransport.Status == 'PCHS':
                PCHSExitToAirsideLane = PCHS_airsidelane_distr.sample()
                yield self.hold(PCHSExitToAirsideLane)
                self.FUtotransport.activate()
                self.enter(AvailableTransportersQueuePCHS)
            elif self.FUtotransport.Status == 'In':
                TransportToFromKC01 = Transport_KC01_distr.sample()
                yield self.hold(TransportToFromKC01)
                self.FUtotransport.activate()
                self.enter(AvailableTransportersQueueKC01)
            elif self.FUtotransport.Status == 'Out':
                TransportToFromKC01 = Transport_KC01_distr.sample()
                yield self.hold(TransportToFromKC01)
                self.FUtotransport.activate()
                self.enter(AvailableTransportersQueueKC01)


class Tractor(sim.Component):
    def setup(self, FUtoRR = None):
        self.FUtoRR = FUtoRR
    def process(self):
        while True:
            yield self.passivate

            if self.FUtoRR.CoolStorage == 'DirectToAirside':            # Check the type of
    ride
                self.FUtoRR.enter(AmbientConditions)
                self.FUtoRR.ExposureTemp.append(tempgenerator.AmbientTemperature)
                self.FUtoRR.ExposureTime.append(env.t_to_datetime(env.now()))
                yield self.hold(5)
                self.FUtoRR.ExposureTemp.append(tempgenerator.AmbientTemperature)
                self.FUtoRR.ExposureTime.append(env.t_to_datetime(env.now()))
                self.FUtoRR.activate()
                self.enter(AvailableTractorsQueue)
            else:
                self.FUtoRR.ExposureTemp.append(tempgenerator.AmbientTemperature)
                self.FUtoRR.ExposureTime.append(env.t_to_datetime(env.now()))
                RRTime = detRRTime(self.FUtoRR.STD)
                yield self.hold(RRTime)
                self.FUtoRR.ExposureTemp.append(tempgenerator.AmbientTemperature)
                self.FUtoRR.ExposureTime.append(env.t_to_datetime(env.now()))
                self.FUtoRR.activate()
```

```
                yield self.hold (RRTime)                                            # Back from aircraft. Same time assumed
                self.enter(AvailableTractorsQueue)


R_A = np.zeros (n)
R_OTP = np.zeros (n)
R_TA = np.zeros (n)
OCCE = np.zeros (n)

No_stored = np.zeros (n)
No_swapped = np.zeros (n)

AvgExposure = list([timedelta(hours = 0)] * n)

for l in tqdm (range(n), desc = "Simulating runs..."):
    """--- INITIALIZATION ---"""
    MissedFlight.clear()
    WasCoolStored.clear()
    WasSwapped.clear()
    with open('trace_log.txt','w') as out:
        env = sim.Environment (time_unit = 'minutes', datetime0 = datetime(2021,1,1), random_seed = randomseed, trace=out)
        env.suppress_trace_linenumbers(Suppress_trace_linenumbers)

        inputgenerator = InputGenerator()
        tempgenerator = TempGenerator()
        pchsentrance = PCHSentrance()
        truckarrival = TruckArrival()
        decisionmodule = DecisionModule()
        controller = Controller()
        kc01 = KC01(COLTemp = COLsetpoint, CRTTemp = CRTsetpoint)
        pchsexit = PCHSexit()


        """--- STATE Definitions ---"""
        ETVbusy = sim.State('ETV busy')
        COLStorageFull = sim.State('COL storage full')
        CRTStorageFull = sim.State('CRT storage full')


        """--- QUEUE Definitions ---"""
        AllFreightUnits = sim.Queue('All ULDs')
        FreightUnitsInProcess = sim.Queue('ULDs in process')
        AllAWBs = sim.Queue('All AWBs')

        storage_PCHS = sim.Queue('PCHS Storage')
        storage_KC01_COL = sim.Queue('KC01 COL Storage', capacity = COLCapacity)
        storage_KC01_CRT = sim.Queue('KC01 CRT Storage', capacity = CRTCapacity)
        HandledFreightUnits = sim.Queue('Handled ULDs')
        entrancequeue = sim.Queue('Entrance Queue')

        AirsideLaneQueue = sim.Queue('Airside lane buffer')

        AmbientConditions = sim.Queue('ULDs in ambient environmental conditions')

        AvailableTransportersQueueKC01 = sim.Queue('Available transporters KC01')
        AvailableTransportersQueuePCHS = sim.Queue('Available transporters PCHS')
        AvailableTractorsQueue = sim.Queue('Available tractors')

        """--- RESOURCE Definitions ---"""
        Transporters = []
        for _ in range(NrTransportersKC01):
            transporter = Transporter()
            Transporters.append(transporter)
            transporter.enter(AvailableTransportersQueueKC01)

        for _ in range(NrTransportersPCHS):
            transporter = Transporter()
            Transporters.append(transporter)
            transporter.enter(AvailableTransportersQueuePCHS)
```

```
    Tractors = []
    for _ in range(NrTractors):
        tractor = Tractor()
        Tractors.append(tractor)
        tractor.enter(AvailableTractorsQueue)

    """--- DISTRIBUTIONS ---"""
    Arrival_distr = sim.External(st.gompertz, c = 14.290598889851587, loc = 5.963957271109827, scale = 432.8146589131147, time_unit='minutes')
    PCHSout_distr = sim.External(st.johnsonsu, a =  -2.1373583494519592, b = 1.4436374905636349, loc = -217.36932520697655, scale = 60.17754370345297, time_unit='minutes')
    PCHSin_KC01in_distr = sim.External(st.johnsonsb, a= 1.1054284233387612,b= 1.043704671177156,loc= 12.352580355238649,scale= 351.4011122382053, time_unit='minutes')
    Transport_KC01_distr = sim.Bounded(sim.External(st.cauchy, loc = 2.5005167413440166, scale = 0.8991652479295638, time_unit='minutes'), lowerbound = 1, upperbound = 20)
    KC01out_distr = sim.External(st.genhyperbolic,p = 1.1177338177432359, a = 0.5137061998096379, b = 0.29790506020349944, loc = 0.04838179962329987, scale = 16.51203203870395, time_unit='minutes' )
    PCHS_airsidelane_distr = sim.Bounded(sim.External(st.lomax, c = 9.62742531642483, loc =  0.9999999999999927, scale = 39.256570439859004, time_unit='minutes'), lowerbound = 0, upperbound = 60)
    Airsidelane_waiting_distr = sim.External(st.invweibull, c = 1.3816423074229707, loc = -3.767827670671733, scale = 14.553388700326352, time_unit='minutes')
    RR_distr = sim.External(st.genhyperbolic, p = -1.4669768102035186, a = 1.115297290476859, b = 1.0908143212514227, loc = 10.573959407968177, scale = 5.6019559650236, time_unit='minutes')


    """--- MONITORS ---"""
    TPTMonitor = sim.Monitor('Total Processing Time')

    """--- ANIMATION ---"""
    env.background_color('30%gray')

    sim.AnimateQueue(FreightUnitsInProcess, x=100, y=550, title='ULDs in Cool Chain', direction='e', id='green')
    sim.AnimateQueue(storage_PCHS, x=100, y=400, title='ULDs in PCHS', direction='e', id='red')
    sim.AnimateQueue(storage_KC01_COL, x=100, y=250, title='ULDs in KC01 COL', direction='e', id='blue')
    sim.AnimateQueue(storage_KC01_CRT, x=100, y=100, title='ULDs in KC01 CRT', direction='e', id='blue')

    sim.AnimateMonitor(storage_KC01_COL.length, x=500, y=550, width=300, height=100, horizontal_scale=5, vertical_scale=5)
    sim.AnimateMonitor(storage_KC01_CRT.length, x=500, y=400, width=300, height=100, horizontal_scale=5, vertical_scale=5)
    sim.AnimateText(text=lambda: TPTMonitor.print_statistics(as_str=True), x=500, y=300, text_anchor='nw', font='narrow', fontsize=9)



    env.animate(Animate)
    env.modelname('Cool chain Digital Twin development study')

    """--- RUN MODEL ---"""
    env.run ()


    """--- PRINT RESULTS ---"""
    #Collecting results ULD level
    ULDRows = []
    ULDPerformance = pd.DataFrame(columns = ['ULD_In','CC1','CC2','CC3','H1','H2','H3','TempAdherence'])
    for uld in HandledFreightUnits:
        newRow = {'ULD_In':uld.name(),'CC1':uld.CCDeadline1,'CC2':uld.CCDeadline2,'CC3':uld.CCDeadline3,
                  'H1':uld.HDeadline1,'H2':uld.HDeadline2,'H3':uld.HDeadline3,'TempAdherence':uld.Exposure}
```

```
            ULDRows.append(newRow)
        ULDPerformance = ULDPerformance.append(ULDRows, ignore_index = True)


        AllExposures = []
        #Collecting results AWB level
        for awb in AllAWBs:
            # TORULDs = []
            ExposureULDs = []
            MissedFlightULDs = []
            for i in awb.myULDs:
                # TORULDs.append((i.TPT - i.TIR))
                ExposureULDs.append(i.Exposure)
                if i in MissedFlight:
                    MissedFlightULDs.append('YES')
                elif i not in MissedFlight:
                    MissedFlightULDs.append('NO')

            if 'YES' in MissedFlightULDs:
                awb.myMissedFlights = 'NOT OK'
            else:
                awb.myMissedFlights = 'OK'

            # if max(TORULDs) > timedelta(hours = 8):              #If max of all ULDs for a
    given AWB is below the commercial promise, then all other ULDs are also OK
            #     awb.myTOR = 'NOT OK'
            # else:
            #     awb.myTOR = 'OK'

            if max(ExposureULDs) > timedelta(hours = 8):
                awb.myExposure = 'NOT OK'
            else:
                awb.myExposure = 'OK'
            AllExposures += ExposureULDs
            # TORULDs.clear()
            ExposureULDs.clear()
            MissedFlightULDs.clear()

        AvgExposure[l] = detAvgTimedelta(AllExposures)

        # TORAdherence = []
        ExposureAdherence = []
        OTP_FAP = []
        for awb in AllAWBs:
            OTP_FAP.append(awb.myMissedFlights)
            # TORAdherence.append(awb.myTOR)
            ExposureAdherence.append(awb.myExposure)

        #Calculating Availability
        Overall_Availability_COL = kc01.COLAvailability['Availability'].mean()
        Overall_Availability_CRT = kc01.CRTAvailability['Availability'].mean()
        R_A[l] = (Overall_Availability_COL + Overall_Availability_CRT) / 2


        #Calculating OTP
        #>ULD level
        # CD1_total = (ULDPerformance['CC1'].value_counts()['OK']) / (ULDPerformance['CC1'].
    value_counts()['OK'] + ULDPerformance['CC1'].value_counts()['NOT OK'])
        # CD2_total = (ULDPerformance['CC2'].value_counts()['OK']) / (ULDPerformance['CC2'].
    value_counts()['OK'] + ULDPerformance['CC2'].value_counts()['NOT OK'])
        # CD3_total = (ULDPerformance['CC3'].value_counts()['OK']) / (ULDPerformance['CC3'].
    value_counts()['OK'] + ULDPerformance['CC3'].value_counts()['NOT OK'])

        # HD1_total = (ULDPerformance['H1'].value_counts()['OK']) / len(HandledFreightUnits)
        # HD2_total = (ULDPerformance['H2'].value_counts()['OK']) / len(HandledFreightUnits)
        # HD3_total = (ULDPerformance['H3'].value_counts()['OK']) / len(HandledFreightUnits)


        #Calculating Temp Adherence
        #>AWB level
```

```
        # Overall_TORAdherence = TORAdherence.count('OK') / (TORAdherence.count('OK') + TORAdherence.count('NOT OK'))
        R_TA[l] = ExposureAdherence.count('OK') / (ExposureAdherence.count('OK') + ExposureAdherence.count('NOT OK'))

        #>AWB level
        FractionMissedFlights = OTP_FAP.count('NOT OK') / len(AllAWBs)

        R_OTP[l] = 1 - FractionMissedFlights


        #Calculating OCCE
        OCCE[l] = (R_A[l] * R_OTP[l] * R_TA[l])*100.00

        No_stored[l] = len(WasCoolStored)
        No_swapped[l] = len(WasSwapped)

R_A_avg = np.sum(R_A) / n
R_OTP_avg = np.sum(R_OTP) / n
R_TA_avg = np.sum(R_TA) / n
OCCE_avg = np.sum(OCCE) / n

No_stored_avg = np.sum(No_stored) / n
No_swapped_avg = np.sum(No_swapped) / n

OverallAvgExposure = np.sum(AvgExposure) / n

if n > 1:
    if EnableDigitalTwin == True:
        title = 'RESULTS - Multiple runs - DT Enabled'
    else:
        title = 'RESULTS - Multiple runs - DT Disabled'
    # deadlines = 'Deadlines'
    availability = 'Cool storage availability'
    otp = 'OTP (FAP)'
    ta = 'Temperature Adherence'
    occe = 'OCCE'

    print('\n')
    print(title.center(100,'%'))
    print('\n')
    # print(deadlines.center(100,'-'))
    # print('Score per deadline over all ULDs:')
    # print('CC Deadline 1:', "{:0.4f}".format(CD1_total))
    # print('CC Deadline 2:', "{:0.4f}".format(CD2_total))
    # print('CC Deadline 3:', "{:0.4f}".format(CD3_total))
    # print('Handling Deadline 1:', "{:0.4f}".format(HD1_total))
    # print('Handling Deadline 2:', "{:0.4f}".format(HD2_total))
    # print('Handling Deadline 3:', "{:0.4f}".format(HD3_total))
    # print('\n')
    print(availability.center(100,'-'))
    print('Availability COL:',"{:0.4f}".format(Overall_Availability_COL))
    print('Availability CRT:',"{:0.4f}".format(Overall_Availability_CRT))
    print('Availability:', "{:0.4f}".format(R_A_avg))
    print('\n')
    print(otp.center(100,'-'))
    print('Missed flights:',"{:0.4f}%".format(FractionMissedFlights*100))
    print('FAP:', "{:0.4f}".format(R_OTP_avg))
    print('\n')
    print(ta.center(100,'-'))
    # print('Current quality measure (TOR vs. Commercial promise):')
    # print('TOR Adherence:', "{:0.4f}".format(Overall_TORAdherence))
    # print('\n')
    print('Utilising ambient and warehouse temp for exposure (Exposure vs. Commercial promise):')
    print('Exposure Adherence:', "{:0.4f}".format(R_TA_avg))
    print('Average exposure:', OverallAvgExposure)
    print('\n')
    print(occe.center(100,'-'))
    print('OCCE = Cool storage availability x OTP x Temperature adherence')
    print('OCCE:', "{:0.2f}%".format(OCCE_avg))
```

```python
else:
    if EnableDigitalTwin == True:
        title = 'RESULTS - Single run - DT Enabled'
    else:
        title = 'RESULTS - Single run - DT Disabled'
    deadlines = 'Deadlines'
    availability = 'Cool storage availability'
    otp = 'OTP (FAP)'
    ta = 'Temperature Adherence'
    occe = 'OCCE'

    print('\n')
    print(title.center(100,'%'))
    print('\n')
    # print(deadlines.center(100,'-'))
    # print('Score per deadline over all ULDs:')
    # print('CC Deadline 1:', "{:0.4f}".format(CD1_total))
    # print('CC Deadline 2:', "{:0.4f}".format(CD2_total))
    # print('CC Deadline 3:', "{:0.4f}".format(CD3_total))
    # print('Handling Deadline 1:', "{:0.4f}".format(HD1_total))
    # print('Handling Deadline 2:', "{:0.4f}".format(HD2_total))
    # print('Handling Deadline 3:', "{:0.4f}".format(HD3_total))
    # print('\n')
    print(availability.center(100,'-'))
    print('Availability COL:',"{:0.4f}".format(Overall_Availability_COL))
    print('Availability CRT:',"{:0.4f}".format(Overall_Availability_CRT))
    print('Availability:', "{:0.4f}".format(R_A_avg))
    print('\n')
    print(otp.center(100,'-'))
    print('Missed flights:',"{:0.4f}%".format(FractionMissedFlights*100))
    print('FAP:', "{:0.4f}".format(R_OTP_avg))
    print('\n')
    print(ta.center(100,'-'))
    # print('Current quality measure (TOR vs. Commercial promise):')
    # print('TOR Adherence:', "{:0.4f}".format(Overall_TORAdherence))
    # print('\n')
    print('Utilising ambient and warehouse temp for exposure (Exposure vs. Commercial promise
    ):')
    print('Exposure Adherence:', "{:0.4f}".format(R_TA_avg))
    print('Average exposure:', OverallAvgExposure)
    print('\n')
    print(occe.center(100,'-'))
    print('OCCE = Cool storage availability x OTP x Temperature adherence')
    print('OCCE:', "{:0.2f}%".format(OCCE_avg))


x_col = kc01.COLAvailability['Timestamp']
y_col = kc01.COLAvailability['#Stored']

x_crt = kc01.CRTAvailability['Timestamp']
y_crt = kc01.CRTAvailability['#Stored']


fig, [ax1, ax2] = plt.subplots(2)


if EnableDigitalTwin == True:
    fig.suptitle('DT Enabled')
else:
    fig.suptitle('DT disabled')
ax1.plot(x_col,y_col, label = 'COL', color = (0,0.6314,0.8706), lw = 0.9)
ax1.axhline(y = COLCapacity, color = (0.4863,0.4980,0.4902))


ax2.plot(x_crt,y_crt, label = 'CRT', color = (0,0.1921,0.2706), lw = 0.9)
ax2.axhline(y = CRTCapacity, color = (0.4863,0.4980,0.4902), label = r'$C_{COL}C_{CRT}$')

plt.rcParams["figure.figsize"] = [12.00, 6]

fig.supxlabel(r'Timestamp [-]')
fig.supylabel('Number of ULDs stored [-]')
```

```
plt.rcParams["figure.figsize"] = [10.00, 6]
plt.figlegend(bbox_to_anchor=(1.02, 0.5), loc='center right', borderaxespad=0)

plt.show()
```

B.2. Digital Twin

```
# -*- coding: utf-8 -*-
"""
Created on Tue Jan  3 20:24:14 2023

@author: klm92970 - Jorrit Sijtsma - 5349664
"""
"""--- Import Packages ---"""
from datetime import datetime, timedelta
# import numpy as np
import pandas as pd
import scipy.stats as st

"""--- Data ---"""
tempinput_dataset = "C://Users//klm92970/OneDrive - Air France KLM/Data/TEMPINPUT_DATA.txt"
dateparse = lambda x: datetime.strptime(x, '%Y-%m-%d %H:%M:%S')
temp_data = pd.read_csv(tempinput_dataset, sep=",",parse_dates=['datetime'], date_parser=
    dateparse)

"""--- Distributions ---"""
meanFromPCHS = st.johnsonsu( a =  -2.1373583494519592, b = 1.4436374905636349, loc =
    -217.36932520697655, scale = 60.17754370345297).mean()
meanWaitPCHS = st.johnsonsb( a= 1.1054284233387612,b= 1.043704671177156,loc=
    12.352580355238649,scale= 351.4011122382053).mean()
meanToCool = 3
meanFromCool = st.genhyperbolic( p = 1.1177338177432359, a = 0.5137061998096379, b =
    0.29790506020349944, loc = 0.04838179962329987, scale = 16.51203203870395).mean()

"""--- Model Parameters ---"""
COLCapacity = 6
CRTCapacity = 28

def DynamicBusinessRuling(CurrentTime, Exposure, SHC, UTB, LTB, AmbientTemp, PCHSTemp, STD,
    COLStored, CRTStored, COLAssigned, CRTAssigned, COLPrevElement, CRTPrevElement):
    """
    Function which encodes the dyanmic business ruling algorithm
    Inputs: Time, ULD characteristics, Exposure, Ambient + PCHS temperature data, STD, Cool
    storage availability + already assigned ULDs
    Output: Cool storage decision (FreightUnit.CoolStorage), cool storage removal
    """
    ULDstatus = 'New'
    # Estimate total exposure for both options
    STD_stored = ''
    Exposure_stored = ''
    SHC_stored = ''


    # Determine the best action based on the minimum exposure
    if SHC == 'COL':
        ExposureKC01, ExposurePCHS = PredictedExposure(Exposure, CurrentTime, UTB, LTB,
    AmbientTemp, PCHSTemp, STD, ULDstatus, STD_stored, Exposure_stored, SHC_stored)
        CurrentAvailability = (COLCapacity - (len(COLStored) + COLAssigned)) / COLCapacity
        if CurrentAvailability >= (1/COLCapacity):
            if ExposurePCHS == timedelta(hours = 0):
                Decision = 'No'
                ULDtoremove = ''
            elif ExposureKC01 <= ExposurePCHS:
                Decision = 'Yes'
                ULDtoremove = ''
            elif ExposureKC01 > ExposurePCHS:
                Decision = 'No'
                ULDtoremove = ''
        elif CurrentAvailability < (1/COLCapacity):
            if ExposurePCHS == timedelta(hours = 0):
```

```
                Decision = 'No'
                ULDtoremove = ''
            elif ExposureKC01 <= ExposurePCHS:
                Decision, ULDtoremove = StorageControl(Exposure, SHC, COLStored, CRTStored, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, COLPrevElement, CRTPrevElement)
            elif ExposureKC01 > ExposurePCHS:
                Decision = 'No'
                ULDtoremove = ''


    elif SHC == 'CRT' or SHC == 'ERT':
        ExposureKC01, ExposurePCHS = PredictedExposure(Exposure, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, ULDstatus, STD_stored, Exposure_stored, SHC_stored)
        CurrentAvailability = (CRTCapacity - (len(CRTStored) + CRTAssigned)) / CRTCapacity
        if CurrentAvailability >= (1/CRTCapacity):
            if ExposurePCHS == timedelta(hours = 0):
                Decision = 'No'
                ULDtoremove = ''
            elif ExposureKC01 <= ExposurePCHS:
                Decision = 'Yes'
                ULDtoremove = ''
            elif ExposureKC01 > ExposurePCHS:
                Decision = 'No'
                ULDtoremove = ''
        elif CurrentAvailability < (1/CRTCapacity):
            if ExposurePCHS == timedelta(hours = 0):
                Decision = 'No'
                ULDtoremove = ''
            elif ExposureKC01 <= ExposurePCHS:
                Decision, ULDtoremove = StorageControl(Exposure, SHC, COLStored, CRTStored, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, COLPrevElement, CRTPrevElement)
            elif ExposureKC01 > ExposurePCHS:
                Decision = 'No'
                ULDtoremove = ''


    return Decision, ULDtoremove


def StorageControl(Exposure, SHC, COLStored, CRTStored, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, COLPrevElement, CRTPrevElement):
    """
    Function which determines the ULD to remove when necessary
    Inputs:
    Output: ULD to remove from storage (if applliccable)
    """
    ULDstatus = 'Storage'
    if SHC == 'COL':
        if len(COLStored) == 0:
            Decision = 'No'
            ULDtoremove = ''
            return Decision, ULDtoremove
        if len(COLPrevElement) == 0:
            min_STD_idx = COLStored["STD"].min()
            min_STD_mask = COLStored["STD"].isin([min_STD_idx])
            min_STD_rows = COLStored[min_STD_mask]
            min_exposure_idx = min_STD_rows["Exposure"].idxmin()
        else:
            mask = list(map(lambda x: not x, COLStored["ULD"].isin(COLPrevElement)))
            COLStored_filtered = COLStored[mask]
            if len(COLStored_filtered) == 0:
                Decision = 'No'
                ULDtoremove = ''
                return Decision, ULDtoremove
            min_STD_idx = COLStored_filtered["STD"].min()
            min_STD_mask = COLStored_filtered["STD"].isin([min_STD_idx])
            min_STD_rows = COLStored_filtered[min_STD_mask]
            min_exposure_idx = min_STD_rows["Exposure"].idxmin()


        STD_stored = COLStored.loc[min_exposure_idx, "STD"]
```

```
            Exposure_stored = COLStored.loc[min_exposure_idx, "Exposure"]
            SHC_stored = COLStored.loc[min_exposure_idx, "SHC"]

    elif SHC == 'CRT' or SHC == 'ERT':
        if len(CRTStored) == 0:
            Decision = 'No'
            ULDtoremove = ''
            return Decision, ULDtoremove
        if len(CRTPrevElement) == 0:
            min_STD_idx = CRTStored["STD"].min()
            min_STD_mask = CRTStored["STD"].isin([min_STD_idx])
            min_STD_rows = CRTStored[min_STD_mask]
            min_exposure_idx = min_STD_rows["Exposure"].idxmin()
        else:
            mask = list(map(lambda x: not x, CRTStored["ULD"].isin(CRTPrevElement)))
            CRTStored_filtered = CRTStored[mask]
            if len(CRTStored_filtered) == 0:
                Decision = 'No'
                ULDtoremove = ''
                return Decision, ULDtoremove
            min_STD_idx = CRTStored_filtered["STD"].min()
            min_STD_mask = CRTStored_filtered["STD"].isin([min_STD_idx])
            min_STD_rows = CRTStored_filtered[min_STD_mask]
            min_exposure_idx = min_STD_rows["Exposure"].idxmin()

        STD_stored = CRTStored.loc[min_exposure_idx, "STD"]
        Exposure_stored = CRTStored.loc[min_exposure_idx, "Exposure"]
        SHC_stored = CRTStored.loc[min_exposure_idx, "SHC"]

    Removal, NoRemoval = PredictedExposure(Exposure, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, ULDstatus, STD_stored, Exposure_stored, SHC_stored)

    if Removal > NoRemoval:
        Decision = 'No'
        ULDtoremove = ''
    elif Removal < NoRemoval:
        Decision = 'Yes'
        if SHC == 'COL':
            ULDtoremove = COLStored.loc[min_exposure_idx, "ULD"]
        elif SHC == 'CRT' or SHC == 'ERT':
            ULDtoremove = CRTStored.loc[min_exposure_idx, "ULD"]
    elif Removal == NoRemoval:
        Decision = 'No'
        ULDtoremove = ''
    return Decision, ULDtoremove

def PredictedExposure(CurrentExposure, CurrentTime, UTB, LTB, AmbientTemp, PCHSTemp, STD, ULDstatus, STD_stored, Exposure_stored, SHC_stored):
    """
    Function which calculates the predicted exposure
    Inputs:
    Output: predicted exposure
    """

    #NewULD
    if ULDstatus == 'New':
        AmbPrediction, PCHSPrediction = PredictedTemp(temp_data, CurrentTime, STD)
        PCHSwait = CurrentTime + timedelta(minutes = meanFromPCHS)
        PCHSremoval = STD - timedelta(minutes = (300 + meanFromPCHS))
        KC01removal = STD - timedelta(minutes = (180 + meanFromCool))
        #Calculate exposure over each period (currentime is head of list, STD is tail of list)
        pchs_wait_idx = ReturnTimeIndex(temp_data, PCHSwait)
        kc01_removal_idx = ReturnTimeIndex(temp_data, KC01removal)
        pchs_removal_idx = ReturnTimeIndex(temp_data, PCHSremoval)
        WaitPCHS = PCHSPrediction.loc[:pchs_wait_idx+1, '[24]']
        FromKC01 = AmbPrediction.loc[kc01_removal_idx+1:, 'T']
        FromPCHS = AmbPrediction.loc[pchs_removal_idx+1:, 'T']
        InPCHS = PCHSPrediction.loc[:pchs_removal_idx-1,'[24]']
        WaitPCHSExposure = timedelta(hours = 0)
        for value in WaitPCHS:
```

```
            if value >= UTB:
                WaitPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                WaitPCHSExposure += timedelta(hours = 1)
        FromKC01Exposure = timedelta(hours = 0)
        for value in FromKC01:
            if value >= UTB:
                FromKC01Exposure += timedelta(hours = 1)
            elif value <= LTB:
                FromKC01Exposure += timedelta(hours = 1)

        #STAYINPCHS:
        FromPCHSExposure = timedelta(hours = 0)
        for value in FromPCHS:
            if value >= UTB:
                FromPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                FromPCHSExposure += timedelta(hours = 1)
        InPCHSExposure = timedelta(hours = 0)
        for value in InPCHS:
            if value >= UTB:
                InPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                InPCHSExposure += timedelta(hours = 1)

        ExposureKC01 = CurrentExposure + WaitPCHSExposure + FromKC01Exposure
        ExposurePCHS = CurrentExposure + InPCHSExposure + FromPCHSExposure
        return ExposureKC01, ExposurePCHS

    elif ULDstatus == 'Storage':
        AmbPrediction, PCHSPrediction = PredictedTemp(temp_data, CurrentTime, STD)
        AmbPredictionStored, PCHSPredictionStored = PredictedTemp(temp_data, CurrentTime, STD_stored)

        PCHSwait = CurrentTime + timedelta(minutes = meanFromPCHS)
        PCHSremoval = STD - timedelta(minutes = (300 + meanFromPCHS))
        KC01removal = STD - timedelta(minutes = (180 + meanFromCool))
        KC01removal_stored = STD_stored - timedelta(minutes = (180 + meanFromCool))
        KC01removal_stored_early = CurrentTime
        #Calculate exposure over each period (currentime is head of list, STD is tail of list)
        pchs_wait_idx = ReturnTimeIndex(temp_data, PCHSwait)
        kc01_removal_idx = ReturnTimeIndex(temp_data, KC01removal)
        kc01stored_removal_idx = ReturnTimeIndex(temp_data, KC01removal_stored)
        kc01stored_removal_early_idx = ReturnTimeIndex(temp_data, KC01removal_stored_early)
        pchs_removal_idx = ReturnTimeIndex(temp_data, PCHSremoval)
        WaitPCHS = PCHSPrediction.loc[:pchs_wait_idx+1, '[24]']
        FromKC01 = AmbPrediction.loc[kc01_removal_idx+1:, 'T']
        FromKC01Stored = AmbPrediction.loc[kc01stored_removal_idx+1:, 'T']
        FromKC01StoredEarly = AmbPrediction.loc[kc01stored_removal_early_idx+1:, 'T']
        FromPCHS = AmbPrediction.loc[pchs_removal_idx+1:, 'T']
        InPCHS = PCHSPrediction.loc[:pchs_removal_idx-1,'[24]']

        WaitPCHSExposure = timedelta(hours = 0)
        for value in WaitPCHS:
            if value >= UTB:
                WaitPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                WaitPCHSExposure += timedelta(hours = 1)

        FromKC01Exposure = timedelta(hours = 0)
        for value in FromKC01:
            if value >= UTB:
                FromKC01Exposure += timedelta(hours = 1)
            elif value <= LTB:
                FromKC01Exposure += timedelta(hours = 1)

        if SHC_stored == 'COL':
            UTB_stored = 8
            LTB_stored = 2
```

```
            FromKC01StoredExposure = timedelta(hours = 0)
            for value in FromKC01Stored:
                if value >= UTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
            FromKC01StoredEarlyExposure = timedelta(hours = 0)
            for value in FromKC01StoredEarly:
                if value >= UTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)

        elif SHC_stored == 'CRT':
            UTB_stored = 25
            LTB_stored = 15

            FromKC01StoredExposure = timedelta(hours = 0)
            for value in FromKC01Stored:
                if value >= UTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
            FromKC01StoredEarlyExposure = timedelta(hours = 0)
            for value in FromKC01StoredEarly:
                if value >= UTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)

        elif SHC_stored == 'ERT':
            UTB_stored = 25
            LTB_stored = 2

            FromKC01StoredExposure = timedelta(hours = 0)
            for value in FromKC01Stored:
                if value >= UTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredExposure += timedelta(hours = 1)
            FromKC01StoredEarlyExposure = timedelta(hours = 0)
            for value in FromKC01StoredEarly:
                if value >= UTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)
                elif value <= LTB_stored:
                    FromKC01StoredEarlyExposure += timedelta(hours = 1)

        #STAYINPCHS:
        FromPCHSExposure = timedelta(hours = 0)
        for value in FromPCHS:
            if value >= UTB:
                FromPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                FromPCHSExposure += timedelta(hours = 1)
        InPCHSExposure = timedelta(hours = 0)
        for value in InPCHS:
            if value >= UTB:
                InPCHSExposure += timedelta(hours = 1)
            elif value <= LTB:
                InPCHSExposure += timedelta(hours = 1)

        ExposureRemoval = (CurrentExposure + WaitPCHSExposure + FromKC01Exposure) + (Exposure_stored + FromKC01StoredEarlyExposure)
        ExposureNoRemoval = (CurrentExposure +InPCHSExposure + FromPCHSExposure) + (Exposure_stored + FromKC01StoredExposure)
        return ExposureRemoval, ExposureNoRemoval

def PredictedTemp(data, CurrentTime, STD):
    time_diff = pd.DataFrame()
    time_diff['time_diff_curr'] = abs((pd.to_datetime(data['datetime']) - CurrentTime))
    time_diff['time_diff_STD'] = abs((pd.to_datetime(data['datetime']) - STD))
```

```
    curr_time_idx = time_diff['time_diff_curr'].idxmin()
    max_time_idx = time_diff['time_diff_STD'].idxmin()


    ambcolumns = ['datetime', 'T']
    pchscolumns = ['datetime','[24]']
    AmbTempPrediction = data.iloc[curr_time_idx:max_time_idx][ambcolumns]
    PCHSTempPrediction = data.iloc[curr_time_idx:max_time_idx][pchscolumns]

    return AmbTempPrediction, PCHSTempPrediction

def ReturnTimeIndex(data, timestamp):
    difference = pd.DataFrame()
    difference['time_diff'] = abs((pd.to_datetime(data['datetime']) - timestamp))
    index = difference['time_diff'].idxmin()
    return index
```


TREPEL
BEST