Philip Santner

Simulation of Logistics for Construction Management

Master's Thesis

Graz University of Technology

Institut für Maschinenbau- und Betriebsinformatik
Head: Univ.-Prof. Dipl-Ing. Dr.techn. Siegfried Vössner

Supervisor: Dr. Nikolaus Furian

Graz, May 2017

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, ____________________ ____________________

Date Signature

Eidesstattliche Erklärung[1]

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am ____________________ ____________________

Datum Unterschrift

[1]Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Foreword and Acknowledgement

First of all I would like to extend all my sincere thanks to some very important persons in my life, who have made this thesis and my studies possible. My biggest thanks go to my parents Anni und Josef who supported me throughout all the years of my education. You were always there for me and you helped me to make my dreams come true. A special thank goes to my brothers Max and Klaus, and to Georg who became like a brother to me in that time. Without the support of my friends and family, none of that would have been possible.

A big thanks to my supervisors form University of Auckland, Dr. Michael O'Sullivan and Associate Professor Cameron Walker. It was a great pleasure to work with such intelligent and creative guys as you are. I really learned a lot from you during that time. I also owe gratitude to Univ.-Prof. Dipl.-Ing. Dr.techn. Siegfried Vössner for the opportunity to write my thesis at his Institute of Engineering and Business Informatics. I want to thank in particular Dipl.Ing. Dr.techn. Nikolaus Furian for enabling me to write the thesis in New Zealand and for his tireless support as my supervisor.

Last but not least, I want to thank Graz University of Technology for supporting my internship by approving the "Research Abroad" scholarship.

Thanks to all of you!

Abstract

The construction sector is faced with increasing competition. Thus, efficiency and agility get more and more important. The first step in optimizing a system is to get a better understanding of the activities and their interconnections. One possibility in order to gather this kind of information is to build a simulation model. This thesis describes the procedure of developing an entire simulation study of construction site supply chain logistics. A high degree of uncertainty is in the nature of every construction project. Permanent unpredictable changes of environmental factors (such as humans, weather or breakdowns) complicate the planning phase tremendously. It is a challenge to find ways in which to deal with this uncertainty in simulation studies. The introduction of a new conceptual modelling framework (HCCM framework) opens doors to new and enriching possibilities to handle these situations. Initially developed for the health care sector, this framework provides tools for representation of even sophisticated systems. The investigated construction site laid a new storm water pipe system in Auckland's central business district in New Zealand. The first phase of this project was to get an overview about the system and to understand the general situation of the problem. In regular meetings with managers of the construction site, the modelling team could get a first impression of the procedures and objectives of the simulation model were defined. The aims of the study were to simulate the logistics processes of the construction project (pipe delivery and soil removal) as well as to analyse its behaviour and to work out potential improvements. Data availability was limited. Thus, one task was to find methods to describe in which ways to deal with a lack of data. The computer simulation was coded with an open source software, namely JaamSim. The final version simulates an entire construction project with the main focus on the storage behaviour of different order policies and numbers of trucks. The conclusion states that a continuous truck order policy accomplishes the best results, compared to orders at a certain time of day. Furthermore, it could be concluded that a minimum of available trucks (one pipe truck and one dump truck) deliver the best outcomes regarding average waiting time and queue length.

Kurzfassung

Wie in der produzierenden Industrie ist auch die Baubranche mit steigendem Wettbewerb konfrontiert. Agilität und Effizienz gewinnen daher immer mehr an Bedeutung. Bevor ein System optimiert werden kann, müssen alle Vorgänge und ihre Verbindungen zueinander bekannt sein. Simulationen bieten die Möglichkeit, das Verhalten genauer zu untersuchen und Rückschlüsse auf das reale System zu schließen. Diese Arbeit beschreibt die Entwicklung einer kompletten Simulationsstudie für Logistikabläufe einer Baustelle. Projekte im Bauwesen sind immer mit hohen Unsicherheiten verbunden. So können unvorhersehbare Ereignisse zu großen Verzögerungen führen. Weitere Einflussgrößen, wie Wetter, Mensch oder Maschinenausfälle, erschweren die Planbarkeit zusätzlich. Die große Herausforderung in Simulationsstudien ist es, mit dieser Unbestimmtheit umzugehen. Einen neuen Ansatz dazu steellt das HCCM-Framework für konzeptionelle Modellierung dar. Ursprünglich für den Gesundheitssektor entwickelt, bietet es neue Möglichkeiten und Werkzeuge selbst komplexe System zu modellieren. Die zu untersuchende Baustelle liegt im zentralen Geschäftsviertel von Auckland, Neuseeland. Im Rahmen des Baues einer U-Bahn werden hier neue Regenwasserkanalisationsrohre verlegt. In der ersten Phase der Studie wurden die generellen Abläufe der Baustelle untersucht, die Problemstellung definiert und Ziele festgelegt. Im Fokus der Untersuchungen standen die Lieferungen der Rohre sowie der Abtransport des Erdmaterials. Neben einer großen Anzahl von unvorhersehbaren Einflussfaktoren war auch Datenmangel ein Herausforderung. Methoden zur Datenaufbereitung mussten untersucht werden, um dennoch ein aussagekräftiges Simulationsmodel zu erstellen zu können. Die Computersimulation wurde mit einem Open-Source Programm erstellt (JaamSim). Die finale Version simulierte verschiedene Bestellstrategien und Anzahl von verfügbaren Lastkraftwägen. Das Ergebnis der Studie zeigt große Vorteile einer kontinuierlichen Bestellpolitik. Im Vergleich zu anderen Methoden konnten hier Verbesserungen im Bereich der Lagerstände aufgezeigt werden. Darüberhinaus wurden auch verschiede Lastkrafwagenkonfigurationen simuliert. Die besten Ergebnisse wurden mit der minimalen Anzahl an verfügbahren Lastkraftwägen erziehlt, ohne dabei die Gesamtprojektdauer zu verzögern.

Contents

List of Figures

Abbreviations

BPMN	Business Process Model and Notation
CM	Conceptual Modelling
DES	Discrete Event Simulation
GPS	Global Positioning System
JIT	Just in Time
NZ	New Zealand
TBM	Tunnel Boring Machine
TPM	Total Productive Maintenance

Chapter 1

Introduction

1.1 Background and Motivation

In the last decades, running a business has gotten more and more difficult. Due to increasing globalization, competition has increased vastly in all sectors. Especially high-wage countries have had to find ways in order to stay competitive. Aside from innovation, continuous improvement has become an established philosophy in the industry. Manufacturing companies have been the pioneers of optimization for years. New methodologies such as Just in Time (JIT) or Lean Management lead the age of production into new spheres. It was not only the manufacturing sector that realised the necessity for optimization, the call for efficiency was also heard in construction management, as increasing prices for resources made their contribution for joining this trend.

In manufacturing there are only marginal changes or rather adaptations of a system for different projects. By comparison, new projects in construction management are faced with differing initial situations. The challenge for construction companies, was how to integrate the new efficient methodologies in their operations. However, in order to optimize a system, all processes and the behaviour of these processes needs to be well known in order to begin optimizing. Furthermore, the big challenge is to integrate findings of one system into a completely new project.

One possible approach is to model and simulate the construction building system in order to gather a better insight. The start of modern construction simulation languages started in 1977 (AbouRizk, 2010b). Since then, great advances especially in information technology were made and an increasing number of complex problems could be modelled and simulated.

Now there is a wide variety of simulation approaches and sophisticated software available. The motivation for this thesis was to develop an entire simulation study of a construction site's supply chain with the use of a new conceptual modelling technique (HCCM framework (Furian et al., 2015)) and an open source software for discrete event simulation (JaamSim).

1.2 Problem Definition

New Zealand (NZ) is faced with a high population growth. Especially Auckland, the biggest city in NZ, is experiencing this trend. In order to improve the tense traffic situation in the central business district Auckland Transport has started to build an underground railway. As part of this project, storm water pipe drainages had to be relocated. At several locations in the city center construction sites were built up where those pipes were "laid". The technique, used for this process is called "pipe jacking".

The construction sites are faced with highly volatile traffic situations and restricted space. An efficient space exploitation as well as a sophisticated supply chain management is necessary in order to guarantee a problem-free realisation of the project. As a result the responsible construction company has to be willing to improve their operational processes. As a first step, the aim is to gather information on the current situation. Furthermore, it is of interest how the system reacts to changes of the supply chain. After all, suggestions for the improvement of future projects should be made.

In order to satisfy these demands, an entire simulation study of the construction site's material supply chain has to be made. Starting with data acquisition and understanding the current situation, a full conceptual model has to be built in consultation with the responsible persons of the construction site. Based on the conceptual model a computer simulation is developed. By running possible alternative scenarios, a better insight of the system should be provided. Findings of the simulation are used for future project planning.

1.3 Goals

In general terms, the purpose of a simulation study is to reduce the resources of a project. Problems can be identified at a very early stage, this in turn reduces efforts for troubleshooting and enables a realization of the project on time. The goal of this thesis is to develop a simulation study which fulfils these requirements. This includes the development of a conceptual model and a computer simulation.

Goals are:

- gathering information about the entire construction site's supply chain with all the processes,
- determination, assessment, and documentation of internal and external influencing factors,
- identification of problems and
- finding possible solutions through simulating alternative scenarios.

So far there are only few simulation studies using the HCCM framework. The major case studies of this framework were completed in the field of health care. One objective of this thesis is to show whether the HCCM framework is also applicable in supply management for construction sites. The same applies to the usage of JaamSim. The discrete event simulation software has excellent benchmark results for its execution speed, but it is not well known for academic research yet. This thesis will show whether the realization of the simulation in this case, and for even more complex cases is possible.

1.4 Structure

The thesis is structured into two main parts. The first part provides the theoretical background for the case study. All basic elements of a simulation study are described in detail. It also gives an introduction of the simulation in the field of construction and supply chain management. The practical part describes the progress of the case study. Results and findings of the simulation are presented subsequently. A discussion about future work and the conclusion are presented at the end of the thesis.

Chapter 2

Theoretical Background

This chapter establishes the theoretical foundation for the thesis and provides the necessary background knowledge for the subsequent case study. Firstly, simulation studies as well as simulation methods (e.g. Discrete Event Simulation) are discussed in general terms. The subsequent part focuses on the elements of a simulation study such as CM, data collection and the computer model. The last part of this chapter deals with supply chain management and applications in the field of construction.

2.1 Simulation Studies: An Overview

This section gives an overview about simulation studies and different simulation approaches in general. It provides the reader with knowledge about basic elements and methods of a simulation study, necessary for the case study.

2.1.1 Simulation Studies

Curiosity is a main characteristics of human beings. Whether because of a thirst for knowledge or just for fun, humans have a need to solve problems and analyse their surroundings. Experimenting was the first and for thousands of years the only technique by which humans could learn about their environment. Although the principle is still the same, technological advances provide vast quantities of new possibilities to satisfy this need. Due to advances of mathematical techniques, researches are able to ask questions, they could not ask before. Thus, more complex problems are solvable. Especially the advent of computers in the 1940s lead to a unprecedented quantity of new methods of analysis. Experiments and simulations could be performed within a virtual environment and enabled applications with a higher level of detail. There are multiple advantages of simulation. Shorter design cycles, increased longevity, higher cost efficiency and the simplicity of modifications are, among others, the reasons why simulations are used in science and engineering more frequently. (Wainer, 2009, p.3-7)

Literature provides different approaches, how to develop successful simulation studies. They differ in the definition of processes and the level of detail. Based on the work of Landry, Malouin, and Oral, 1983, Brooks and Robinson, 2000 defined four major stages and processes within a simulation study.

- A conceptual model: describes the simulation model based on the real world problem situation
- A computer model: the implementation of the conceptual model in the computer
- Solutions and/or understanding: Results of the computer model
- Real world (problem): the improvement of the real world problem by using the findings of the computer model

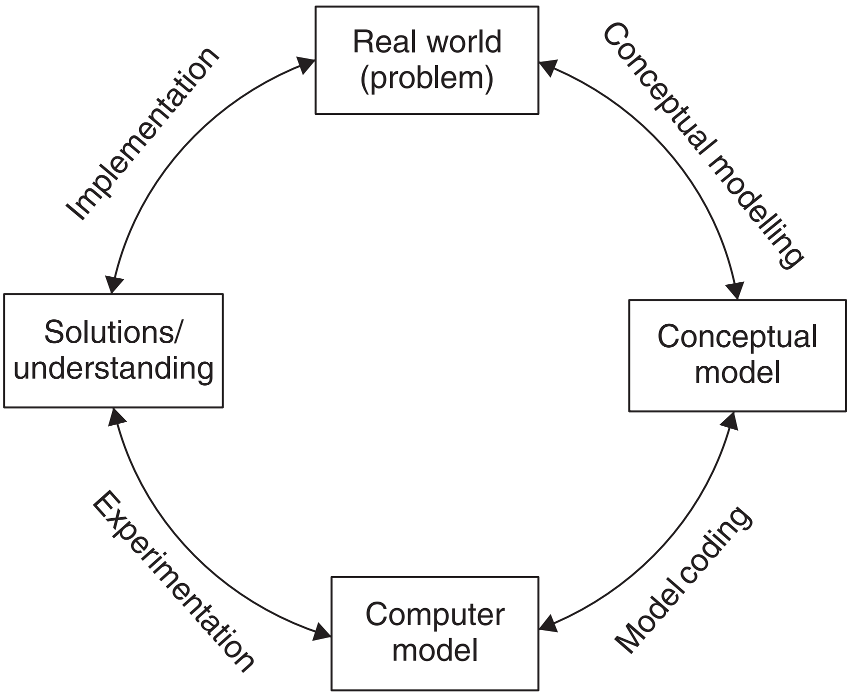


Figure 2.1: Key stages and process within simulation studies (Brooks and Robinson, 2000)

The main stages interact with each other. The processes between those steps are displayed as arrows between these steps.

The aim of the first stage is developing an understanding of the real world situation and the description of the problem. Furthermore, the modelling objectives have to be defined. Gathered information such as model inputs, outputs and model content are defined in the conceptual model. The data gathering and interpretation is one important part in this process. The subject of data gathering is described in greater detail in Chapter 2.4. After defining the first conceptual model, the information is used to code a computer model. The decision of which software solution is taken depends on the conditions of the real world problem and the collected data. The section 2.5 delivers an overview about computer simulation and software solutions. In order to get a better understanding of the practical problems, experiments with the computer simulation models are performed. By varying input factors, information about the system's behaviour is collected. The aim of this stage is to look whether the simulation is valid or not and how the conceptual and computer model have to be adopted to reach the necessary scientific validity. In the fourth stage the findings of the simulation experiments are implemented in to practice. This can be done by putting a particular solution into practice in real time, such as changing processes or implementing, not just the solutions of the model, but rather the model itself. An example could be simulating alternative production schedules for manufacturer at the beginning of each week. A third way of implementation is if the simulation helps stakeholders to better understand the system and hence, are able to make better decisions. (Robinson, 2004, p.51-54)

2.1.2 Simulation Approaches

Banks et al., 2004, p.3 defines simulation as follows:

"A simulation is the imitation of the operation of a real-world process or system over time. Whether done by hand or on a computer, simulation involves the generation of an artificial history of a system and the observation of that artificial history to draw inferences concerning the operating characteristics of the real system."

The behaviour of a real world system is described by the simulation model. A simulation model can be classified into three dimensions (Law and Kelton, 1991, p.6ff):

- Static vs. Dynamic Simulation Model
- Deterministic vs. Stochastic Simulation Model
- Continuous vs. Discrete Simulation Model

A simulation model can be defined as static, when time is not a factor. Alternative there are dynamic simulation models which represent systems with variable factors over time. The second dimension describes if a system includes probabilities or randomness. Systems without any randomness are defined as deterministic because the output is determined just on the basis of the input factors. However, almost all real world processes have a certain randomness and hence, have to be modelled as a stochastic processes. A system is defined as continuous when changes within the system occur continuously with respect to time. In contrast, a system with instantaneous changes at certain points in time is discrete. Although the majority of real world systems consist of both continuous and discrete parts, the modeller has to decide which type represents the situation best. (Law and Kelton, 1991, p.6ff)

The simulation models, this thesis deals with are dynamic, stochastic and discrete. This type of simulation model is called Discrete-Event Simulation (DES) and is described more detailed in the section 2.2.

Another perception on simulation approaches is given by Borshchev and Filippov, 2004. Figure 2.2 classifies approaches in their level of abstraction. System Dynamics is a modelling paradigm which deals with aggregates and global causal dependencies. In contrast, Dynamic Systems deal with simulation at the micro level, where individual objectives with exact physical properties are simulated. Both are mainly continuous models. Agent Based and Discrete Event Modelling are rather discrete approaches. Whereas Agent Based is essentially decentralized in its structure, Discrete Event uses a global entity processing algorithm. The usage of the Agent Based approach are appropriate, for example, in simulations of all kind in systems with human interactions, such as modelling consumer behaviour, social networks or traffic behaviour. The core of Agent Based simulation are active objects with individual behaviour rules. (Borshchev and Filippov, 2004, p.2ff)

For the sake of completeness, it needs to be mentioned that there are already simulation models which combine continuous and discrete approaches. Rabelo et al., 2005 developed a hybrid approach that combines System Dynamics with DES. In a case study for enterprise simulation the impact of production decisions on enterprise-level performance measures is examined. They conclude that with DES alone the impact cannot be captured

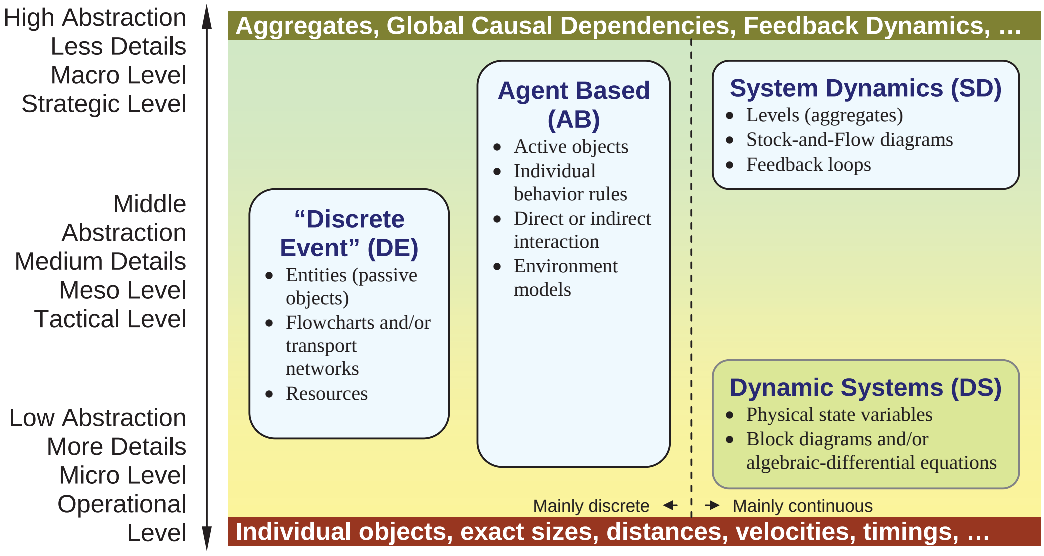


Figure 2.2: Paradigms (Approaches) in Simulation Modelling on Abstraction Level Scale (Borshchev and Filippov, 2004, p.3)

sufficiently. The entire enterprise is modelled with a System Dynamics approach. DES is integrated in this system for selected (mainly operational) parts. Evaluations of this method show valuable results and a high potential for future studies. (Rabelo et al., 2005)

Since, for this thesis, mainly discrete modelling is of importance, the following parts observe this technique in greater detail. In order to specify the term of discrete behaviour the following figure classifies the different modelling techniques by their time bases and state variables.

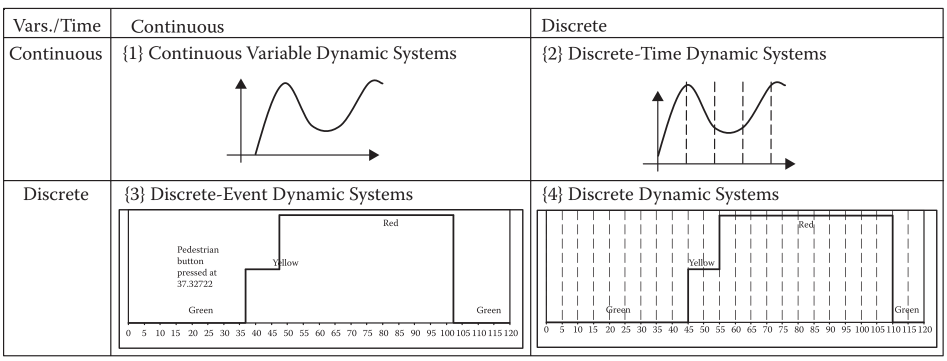


Figure 2.3: Classification according the representation of time bases/state variables (Wainer, 2009, p.16)

The vertical classification in figure 2.3 relates to the values of the state variables. A variable obtains its value either from a continuous dataset or from a discrete set, represented as integer numbers. The methods can be distinguished into two categories (column); continuous (time is a real number) and discrete time techniques (time is represented as an integer number). An example for Continuous Variable Dynamic Systems (1) is the behaviour of a room's temperature. In this case, a sensor which measures the temperature at fixed periods can be described by Discrete-Time Dynamic System (2). A system with a Discrete-Event Dynamic System behaviour (3) would be the arrival of students in the classroom. In the same example, an hourly alarm in the classroom would be represented by a Discrete Dynamic System (4). (Wainer, 2009, p.16)

Before the Discrete Event Simulation technique is explained in greater detail, one further distinction in discrete simulation needs to be mentioned. In general, two approaches can be applied; Discrete Event Simulation and Time Driven Simulation. Those two methods differ in the way that variables change their values. In Time Driven Simulations the clock advances in constant time units. Changes of variables occur only in the subsequent time step (see figure 2.4). This method is less widely used in discrete systems but has to be mentioned for the sake of completeness. On the other hand, In Discrete Event Simulations the progress of changes is not defined by constant time steps, but by a list of events. In other words, this list defines all changes within the system and periods of inactivity can be skipped.

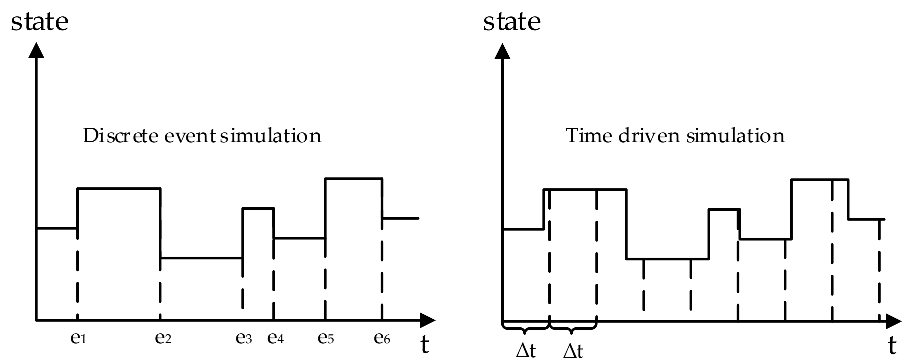


Figure 2.4: Discrete event simulation vs. time driven simulation (Hedtstück, 2013, p.22)

2.2 Discrete Event Simulation

The objective of this chapter is to describe basic functions and the main components of Discrete Event Simulation (DES).

The core of DES is the event list. Whenever an event occurs it effects a change in the state of a system. This implies that the system can only change at a countable number of points in time. Since there are no changes of the system between events, this time can be skipped. Basically DES can be done by means of hand calculations. However, real-world problems have to cope with a big amount of data which can be manipulated and stored on a digital computer. (Law and Kelton, 1991, p.7ff)

The difference to other simulation techniques is in that DES models are solved in a numerical way. Whereas analytical approaches solve a model by solving certain equations, DES rather "runs" models than solve them. Also Banks et al., 2004 suggests the usage of computers to run the simulations and further states that simulating small models manually can bring useful insights of the system. (Banks et al., 2004, p.12)

The general procedure of DES is defined by a scheduled event list. This list consists of all events, associated with their specific time of occurrence. At the beginning of every simulation the actual state at time zero has to be defined. This process is called initializing. In case of randomness within a model a random variate generator assigns the corresponding values to the events. Subsequently, the scheduled event list is generated and all entries sorted with their occurrence time in increasing order. At the same time, the first event e_1 is the triggering event during the initial state with the occurrence time t_1. The actual simulation procedure is defined by six continuously repeating steps. Figure 2.5 shows the scheme of event scheduling in computer simulations.

The scheduled event list is defined by

$$L = \{(e_k, t_k)\} \ , \ k = 1, 2, ..., m_l$$

where m_l is the set of feasible events of the simulation. A change of a state is represented by the state transition function

$$f(x, e')$$

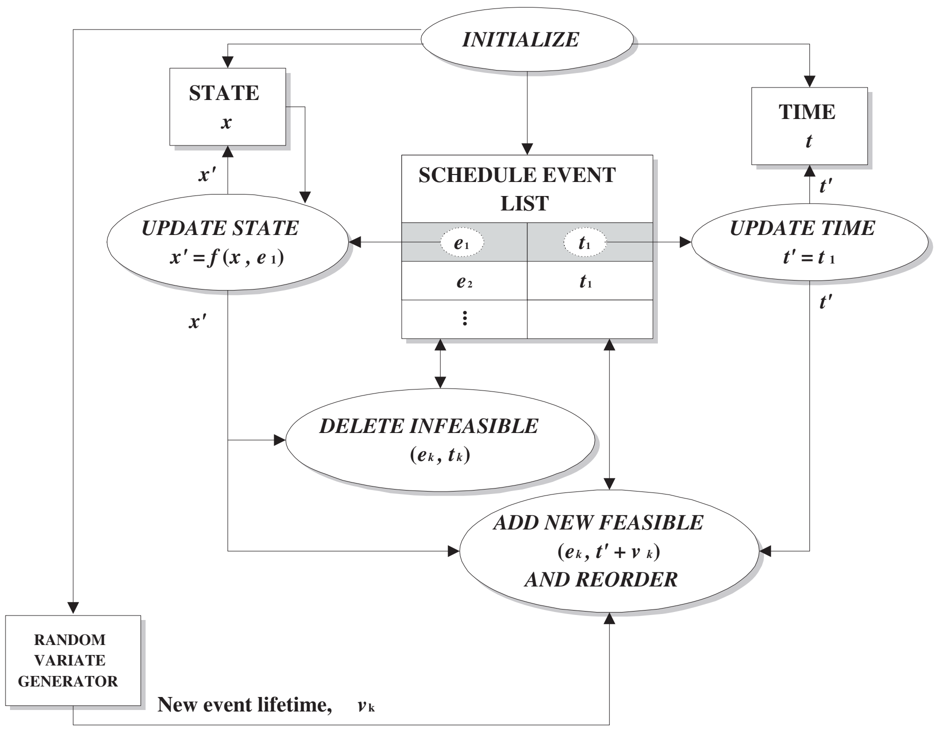


Figure 2.5: The even scheduling scheme in computer simulation (Cassandras and Lafortune, 1999, p.561)

where x defines the current stage and e' the triggering event.

- *Step 1:* The first entry $(e_1,\ t_1)$ is removed from the scheduled event list.
- *Step 2:* The simulation time is updated by proceeding the next event time (t_1).
- *Step 3:* The transition function $x' = f(x, e')$ defines a new state of the system.
- *Step 4:* Delete any entries (e_k, t_k) from the scheduled event list corresponding to events which are infeasible in the new state.
- *Step 5:* Depending on the state, add any feasible event to the scheduled event list which is not already scheduled (this includes the triggering event removed in *Step 1*).
- *Step 6:* Based on a smallest-scheduled-time-first scheme, reorder the new scheduled event list.

2.3 Conceptual Modelling

There is a rapidly growing literature on conceptual modelling over the last decades. Since simulations are getting more and more complex, a well considered phase of preparation takes on greater significance. Moreover, new frameworks of CM are developed which attempt handling with highly complex problems.

The term 'conceptual model' is used in many different areas. In this thesis, the conceptual model and the process of developing the model, Conceptual Modelling (CM), refer to the area of simulation.

There are numerous different definitions of a conceptual model but the following definition by Robinson, 2008a has become established in literature;

'The conceptual model is a non-software specific description of the computer simulation model (that will be, is or has been developed), describing the objectives, inputs, outputs, content, assumptions and simplifications of the model.' (Robinson, 2008a)

To put it in another way, the conceptual model describes the transformation from a complex real-world problem into a logical virtual problem.

In former times CM was faced with a lack of attention. It was seen more as 'art' than 'science' and therefore less researchers focused on this topic. This is one reason why conceptional modelling is a relatively young research field. However, the high potential of this field was recognized in the 2000s and researchers attempted to bring more clarity in this area. (Robinson, 2008a, p.278-279)

The first work on model base concepts was done by Zeigler, 1987. This paper describes the first development of discrete event simulation schemes with modular and hierarchical specifications. Subsequently reputable researches clarified the definition and objective of CM. Balci, 1994 defines the simulation study as a life cycle and the CM as one important part of it. Nance et al., 1994 describes the CM as a model which only exists in the mind of the modeller, the actual documented model which can be compared against the system is named Communicative Model. Further researches of Pace, 2000 and Lacy, Youngblodd, and Might, 2001 develop the interpretation of CM. A more recent work was done by Balci and Ormsby, 2007. Although the author provides new views on the development and structure of CM the work is based on large scale simulations in the defence sector and thus differs to business orientated models where less domain experts and developer are involved. Beside Birta and Arbez, 2007b and Karagöz and Demirörs, 2011 the most recognized definition of CM is provided by Robinson, 2008a. The author delivers a framework based on existing literature which meets the four requirements of a conceptual model: validity, utility, credibility and feasibility. However, due to limits of this approach Furian et al., 2015 introduces the Hierarchical Control Conceptual Modelling (HCCM) framework which represents the current state of science.

In general, a simulation model consists of three layers; Simulation Conceptual Model, Simulation Model Design and Simulation Model Implementation/Programming. The Simulation Conceptual Model represents the highest layer of abstraction and at the same time the base of Simulation Model Design and Simulation Model Implementation/Programming layer. (Balci and Ormsby, 2007, p.176)

The latter two layers are considered more detailed in the section 2.5

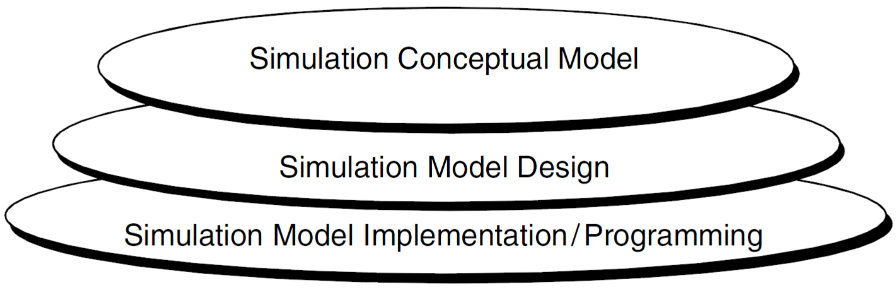


Figure 2.6: Layers of Simulation Model Abstraction (Balci and Ormsby, 2007, p.176)

Figure 2.7 shows a good overview about objectives in CM, its connections and interactions within the system. As mentioned above, CM transforms a real world problem into a simplified representation. In the first step information about the problem is gathered in order to get a representative description of the system. Since not all the information can be collected in the first place, assumptions need to be made and documented. Reasons why the knowledge acquisition is limited are (Kotiadis and Robinson, 2008, p.952ff):

- In some cases the system does not even exist yet or states of the system cannot be observed.

- Observation of the real system and collecting data are prone to error, especially when data is manually recorded.
- Observers are not able to record each smallest interaction in the system. Thus, an observation is incomplete.
- Observations are subjective snapshots of the system. Observers have different perceptions of a problem and this leads to varying results further down the line.

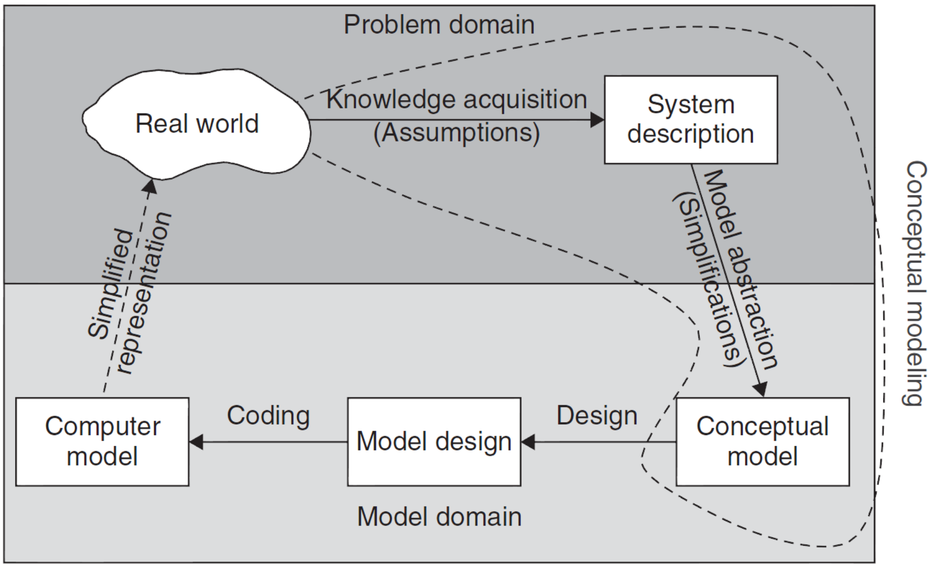


Figure 2.7: Artifacts of Conceptual Modelling (Kotiadis and Robinson, 2008, p.952)

The modeller decides about simplifications after consultation with the problem owner. All changes, assumptions as well as simplifications, are assessed and documented in case of alteration in the system problem or for reusage purposes. In general, CM must not be understood as a linear process but rather as circle in which the conceptual model is evaluated and developed steadily with the real world situation.

Modellers are often faced with problems within simulation modelling. These are (Balci and Ormsby, 2007, p.176):

- communication between simulation developers, stakeholders, analysts and managers
- high complexity within large-scale simulation model development
- coping multidisciplinary knowledge
- re-usage of simulation models or sub-models from an earlier stage and
- verification, validation and certification of simulation models.

Especially in large-scale projects a high number of stakeholders are involved. Each group has different levels of expertise and varying background and interests. This diversity leads to significant challenges in communication. Furthermore, the communication efforts increase strongly with the scale and complexity of the project. This can be very time consuming and challenging in the first place. However, a CM is not used just once ideally. Hence, the investment at this stage is paid off after reusing the CM. The difficulty in reusing earlier developed simulations at the implementation level is attributed to the use of different simulation specifications such as varying programming languages, hardware platforms, operating systems or software solutions. Thus, a CM has to be created, used and maintained frequently in order to cope these problems at an early stage. (Balci and Ormsby, 2007, p.176-177)

2.3.1 Simulation Accuracy and Complexity

Since the quality of the conceptual model affects the validity of the simulation tremendously, it is vitally important to focus on a well-designed model. But trying to achieve a highly meaningful model often leads to over-complex models. Simplicity provides a number of advantages. Figure 2.8 displays the 80/20 rule. With just 20% of the complexity, an average accuracy of 80% can be gained. Furthermore, it shows that 100% accuracy can never be reached, since it is not possible to integrate all influencing parameters of a system in the model. (Robinson, 2004, p.68)

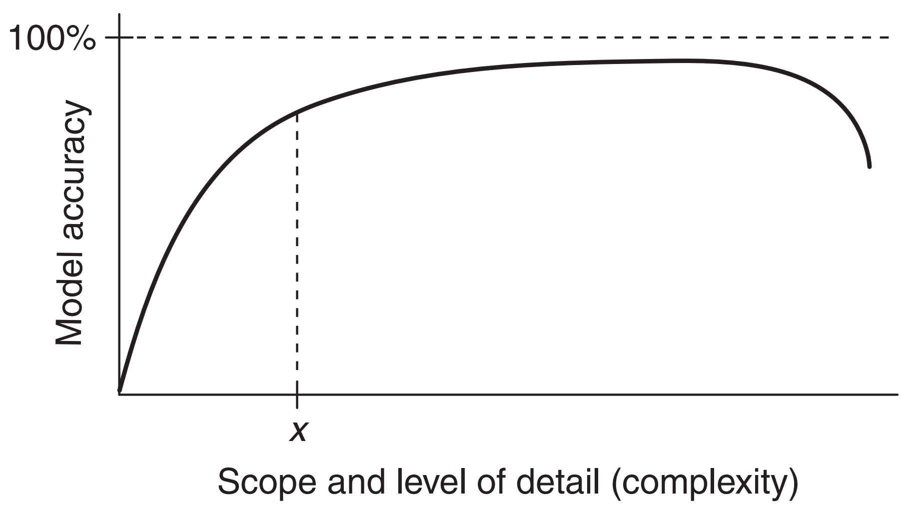


Figure 2.8: Simulation model complexity and accuracy (Robinson, 1994)

A model abstraction, including simplifications, is also necessary because not every detail is relevant for the simulation of the system. Kotiadis and Robinson, 2008 states five main advantages of simple models (Kotiadis and Robinson, 2008, p.953):

- faster development
- more flexibility
- less data required
- shorter running times
- simpler interpretation of results and understanding of the model

'All models are wrong, but some are useful.'
Box, 1979

2.3.2 Frameworks

This subsection provides a brief overview of different frameworks for CM. A framework provides steps and tools in order to support the modeller in developing a conceptual model. Literature gives just a few frameworks and some were developed just for one specific field of application. However, an ideal framework should be applicable universally and for any type of system or initial problem.

Frameworks with the most attention in literature are:

- Frameworks of Karagöz and Demirörs, 2011: Base Object Model (BOM), Conceptual Model Development Tool (KAMA), Conceptual Models of the Mission Space (CMMS),Defence Conceptual Modelling Framework (DCMF) and Federation Development and Execution Process (FEDEP).
- The PartiSim framework, (Tako et al. 2010)
- Conceptual modelling framework for manufacturing, (van der Zee 2007)
- The Robinson framework, (Robinson, 2008b)
- The ABCmod conceptual modelling framework, (Birta and Arbez, 2007b)
- Hierarchical Control Conceptual Modelling framework, (Furian et al., 2015)

Three frameworks are the domain of interest for this thesis; The Robinson framework, the ABCmod framework, and the Hierarchical Control Conceptual Modelling(HCCM) framework. The latter of these three is used in the case study and is based on the other two frameworks. After describing general notations and techniques of CM frameworks, the Robinson and the ABCmod framework are outlined briefly in order to provide the necessary background knowledge for the subsequent HCCM framework.

General Notations and Techniques

In general, the first phase of CM is to understand the defining context and the problem situation. Determining the objectives of the simulation system is part of the context definition. In order to satisfy the objectives, relevant information sources have to be identified and associated assumptions and constraints need to be made. Both, changes and the history of changes must be well documented. The subsequent phase is developing the content. This includes defining inputs, outputs, tasks, interactions, entities and relationships among the system. The objective of this phase is to get an understanding of the system. A clear documentation with text, tables and diagrams is the basis for further work. (Karagöz and Demirörs, 2011, p.182ff)

Each framework has its own rules and methods in notation which may vary widely. Literature delivers a wide range of different methods and notations used in CM. Besides UML (Unified Modelling Language), a frequently used object orientated modelling language for design of software, many other notations such as process flow diagrams, event graphs and activity cycle diagrams are recommended within the framework. (Karagöz and Demirörs, 2011, p.182ff)

In addition to this methods, also free text description approaches are used in several frameworks to define the system. However, this notation causes recurrent and ambiguous definition and may lead to misunderstanding of the problem by the modeller. (Sudnikovich et al., 2004, p.670)

The Robinson Framework

Robinson is one of the leading researchers in this field of research. With his framework he provided a new thinking about CM. It is based on 20 years of experience in developing simulation models. As one of the first frameworks it delivers a step-by-step instruction in order to develop a conceptual model for DES based on the essential requirements; validity, credibility, utility and feasibility. (Robinson, 2008b, p.291)

The approach of Robinson identifies five key activities in developing a conceptual model (Robinson, 2008b, p.291):

- understanding the problem situation
- determining project objectives
- identify model responses (outputs)
- identify model experimental factors (inputs)
- defining the content of the model (level of detail and scope) as well as assumptions and simplifications

As it can be seen, these steps are very similar to the general phases as discussed in section 2.3.2. But here, each activity is documented by summarizing the decision making in tables. This graphical presentation supports the modeller in the phase of creating a model, as well as being used to aid the communication among stakeholders in the understanding of the problem. The following figure shows the process of CM. It should be mentioned that the problem situation may not be part of the conceptual model but a part of the CM process.

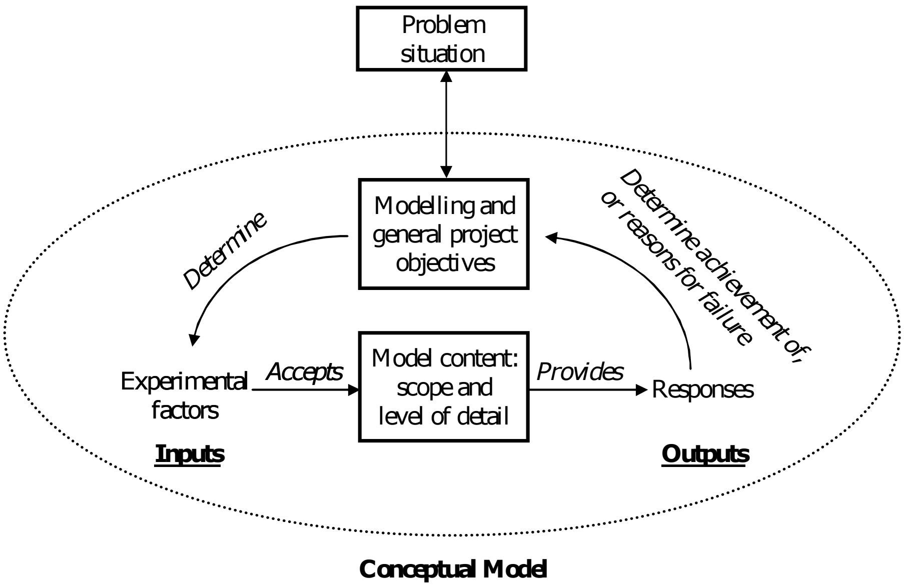


Figure 2.9: A framework for designing the conceptual model (Robinson, 2008b, p.292)

Robinson (2008b) states that a conceptual model can be defined in terms of four main types of components: Entities, Activities, Queues and Resources. However, the author also says that a framework can be extended with additional component types such as transporters or elements of continuous processing systems (e.g. belt conveyors).

ABCmod framework

The ABCmod (Activity-Based Conceptual modelling) framework describes a procedure for discrete-event simulation models identifying its components and relationships. It is based on the usual practice in CM, however, Birta and Arbez, 2007a provide different approaches in structuring and developing the conceptual model.

The ABCmod framework differentiate into two main constituents; model behaviour and model structure. The model structure is divided into entities (resources, consumers, queues and groups), attributes and state variables. The model behaviour consists of events, actions and activities within a system. The CM development is a two-stage process. In the initial stage, a preliminary perspective of the model is formed (high level) and provides the basis for the the second stage in which all detailed information needed for the modelling team, is defined (detailed level). (Birta and Arbez, 2007b, p.110ff)

ABCmod has a very high degree of adaptability, generality and precision at the same time. It provides the modeller a wide range of tools in order to develop a meaningful conceptual model. The authors state that the framework has space for extensions and is not intended to be defined rigidly. (Birta and Arbez, 2007b, p.101)

2.3.3 HCCM Framework

The motivation of Furian et al., 2015 to develop a new framework for CM was the lack essential features of former frameworks. The author states that frameworks such as the Robinson and the ABCmod framework deliver a good guidance for the CM process, however, they do not provide concepts for modelling the system's behaviour on a global level. This is needed especially when dealing with control polices in a system such as dispatching. One example for such policies are decision making for staff members in health care. Since it is not clear whether clinical personal serve a queue or queue themselves up, a simulation model represented by queues may not deliver satisfying results. Both, Robinson, 2008a and Birta and Arbez, 2007a developed their frameworks with the assumption that DES can be represented best with queueing systems. The HCCM breaks with this assumption by introducing *control units*. They represent rules how entities are governed within the system without the need of queues. Defining such control units needs the same care as other components of the conceptual model do. (Furian et al., 2015, p.4ff)

The Structure of the HCCM Framework

The HCCM framework combines essential parts of two frameworks; the Robinson framework and the ABCmod framework. Based on major steps of Robinson, 2008a, the HCCM model divides into structural and behavioural components as recommended in the ABCmod framework. Furthermore, control structures and associated rules (control units) are added. The HCCM framework process, as illustrated in figure 2.10, is divided into 4 phases.

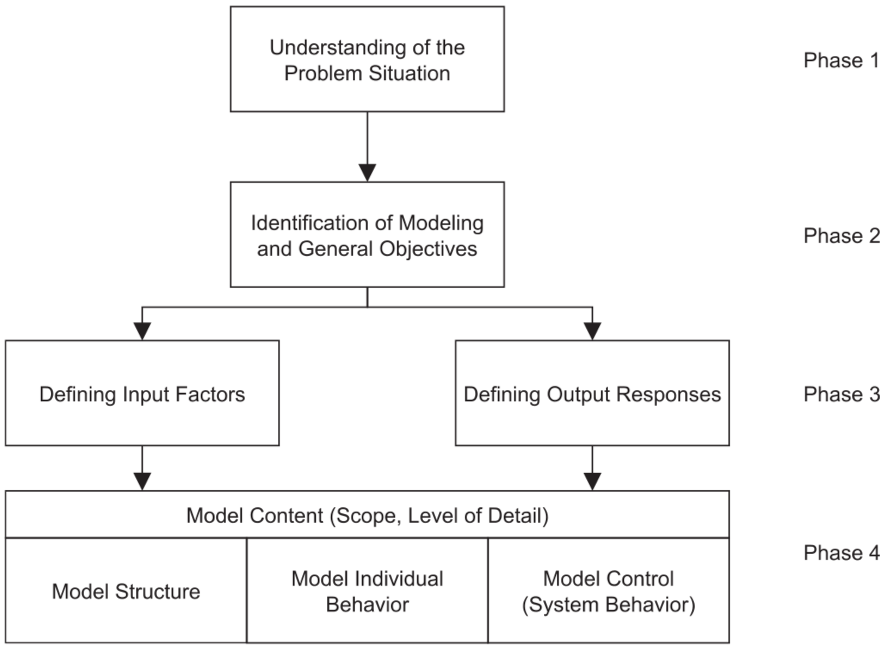


Figure 2.10: Structure of HCCM framework (Furian et al., 2015, p.8)

Phase 1: Understanding the Problem Situation

The first phase in the HCCM is the understanding of the system and the problem situation. As Balci and Ormsby, 2007 and Robinson, 2008b pointed out, at this stage several challenges may arise. Various stakeholders may understand the behaviour and problems of a system differently. Thus it is vitally important to figure out who is able to deliver the best information and data. Furthermore it is crucial to ask the proper questions in order to get a full understanding of the system and its behaviour. All gathered information needs to be well documented. Furian et al., 2015 recommends using formal problem structuring methods. The result of this step is an informal document providing an overall understanding of the system and problem situation. It includes a complete description of the system's situation as well as a documentation of made assumptions in textual form. (Furian et al., 2015, p.8)

Phase 2: Identification of Objectives

Two kinds of objectives are distinguished by Robinson, 2008a and used within the HCCM framework; modelling objectives and general objectives. Modelling objectives define overall aims of the organization. They describe states which can be achieved by using and developing the model. An example for a modelling objective could be increasing the entire productivity of a system by 10 percent. General objectives are determined by the nature of the simulation project and include general requirements for the modelling process by itself such as re-usability, run time, flexibility to changes or visualization requirements. (Furian et al., 2015, p.8)

Phase 3: Input Factors and Outputs

Responses of a simulation are called outputs and define whether the modelling objectives are met or not. They are used to asses the outcome of the simulation and can give reasons why objectives have met or not. Outputs can be streamed data such as time series or aggregated numerical values such as minimum, maximum or variance values. They can be displayed in either tabular or graphical way and have to correspond with the modelling objectives. The inputs define the flexibility of a system. These experimental factors can be changed in order to influence the simulation and thus the output values. However, input factors can contain not only single measures but also policies (e.g. for different dispatching strategies). These policies define the system behaviour and have to be correctly included into the conceptual model. (Furian et al., 2015, p.8ff)

Phase 4: Model Content (Scope, Level of Detail)

The last phase outlines the process of defining the content of the model. Based on Birta and Arbez, 2007b it is structured into Model Structure, Model Individual Behaviour and extended by the Model Control respectively System Behaviour. The objective of this last step is to not only define the content but also point out the boundaries of the model (Scope and Level of Detail).

Model Structure After gathering all necessary data about the system, this information needs to be structured. Any DES model can be described by entities and their aggregations (e.g. queues or groups). However, in HCCM aggregations are defined in ways of *control units*, see paragraph 2.3.3.
The HCCM framework distinguishes only between two different types of entity structures; active and passive. Active entities have an active role within the system and thus a specific behaviour such as consuming, providing resources or changing their roles. Passive entities in turn do not have any active behaviour. Firstly, all entities in the system have to be collected in order to find an appropriate structure. After assessment of all entities the modeller defines the scope and the level of detail by determining which entities to include and further how they are specified. The author makes use of UML (Unified Modelling Language) class diagrams to document the entity structure. Furthermore Furian et al., 2015 proposes to represent the system in an informal graphical way where entities with fixed locations picture the structure of the model and active and moveable entities represent the flow of the model. Further displaying techniques, which help stakeholders to better understand the model, are recommended and their usage is in the discretion of the modeller. (Furian et al., 2015, p.9)

Model Individual Behaviour After defining the model structure the individual behaviour of entities has to be identified. Especially in this process it is vitally important to focus to the priorly defined scope and the level of detail of the model. The modeller has to decide whether an entity's behaviour or part of it, has effects on the validity of the model and therefore whether to include or to exclude (level of detail). Furthermore, only actions of entities within the boundaries need to be considered. The HCCM framework suggests using visual representations such as activity cycle diagrams to show flows of entities within a system. Other methods to show this behaviour can be flow charts, event graphs, business process diagrams, UML or sequence diagrams. It is the modeller's decision which package to choose and mainly depends on the project data and personal preferences. Additionally to a visual documentation, detailed information about the activists need to be reported (e.g. in tabular form). This includes all attributes of entities, state changes, participating entities or information about request made for a certain activity. (Furian et al., 2015, p.9-10)

Model Control (System Behaviour) In order to describe the behaviour of a system without viewing them as queueing systems, Furian et al., 2015 introduced *control units*

in the modelling process. This element represents the core of the HCCM framework and extends the conceptual model with an advanced control mechanism. The control view of the framework is built up as a hierarchical structure with three key elements; control units, activities and events. Control units are defined by a set of rules or strategies and are able to trigger and manage the behaviour of activities and events. Control structures have to be chosen very carefully. Whereas many control units lead to a high complexity of the model, choosing too less control units results in rising condition complexity.
Within some systems, entities have to change their roles and act as a resource for one task and as a customer for a different task. The challenge of former frameworks with entity aggregation structures (queues and groups) was, how to integrate these actions into the conceptual model. Within HCCM, control units manage these tasks and, thus, replace queues and groups.
Control units have to be determined by a well selected set of rules and scenarios in order to handle complex interactions of entities and their requests. In the last step the system's control structure and associated rules have to be identified. In order to include all important rules and decisions within the scope, the process of decision making has to be understood. After gaining all necessary information, Furian et al., 2015 suggests to visualize the organization of control in a tree structure. Finally, the control units are specified detailed and rule-sets for each control unit are outlined. (Furian et al., 2015, p.10ff)

This four phase method has to be viewed, not as a linear sequence of methods and techniques, but rather as a cycle of continuous improvement. All phases interact and influence each other. Findings at subsequent phases can change the view of the initial situation and vice versa. Thus, an accurate documentation of CM is crucial. All assumptions and simplifications of the conceptual model have to be reported in a structured way.

The HCCM framework provides several advantages compared to former frameworks. More complex scenarios can be regarded within the control units by using a centralized description and selection policies. Furian et al., 2015 shows advantages especially for systems where entities change their roles such as for staff members in health care. However, one objective of this thesis is to see whether the application of this framework is also possible or advisable for other areas .

2.4 Data Collection and Analysis

In any simulation study, the collection and analysis of data is one of the most important parts. Since the quality of the simulation outcome mainly depends on their input, special attention is necessary when dealing with this topic. Because of this relevance, Trybula, 1995 states that the process of data gathering and validation can take up to 40% of the entire project time. However, in practice some questions have to be answered quickly which makes it necessary to develop a good simulation model in a short period of time with less or poor data. This chapter provides the reader a base understanding about data collection, analysis and the validation process. Furthermore approaches are pointed out on how to deal with a lack of data or "low quality" data.

The data gathering process is faced with four common difficulties. (Law, 2005, p.8)

- Data do not represent the information, which one wants to model.
- Data formates or types are not appropriate.
- Data includes recording, measuring or rounding errors.
- Data is "biased" due to self-interest of involved parties.

In order to guarantee the validity of the gathered data, it needs to be assured that the logic of the model is correct and second, that appropriate data is used. Law, 2005 describes two basic principles a modeller has to follow in order to gather proper data for the model. Firstly, the modeller has to communicate with people who provide the data requirements such as data type, amount, why it is needed, format and the conditions under which the data needs to be collected. Secondly, a good understanding of the underlying data gathering process is necessary in order to assess the validity of the data. (Law, 2005, p.7)

Figure 2.11 shows the data identification and collection process which is triggered whenever data is needed for the simulation project. It describes the basic steps in data gathering. Key elements are rules for treating available data (e.g. clean, analyse and verify data) and not available data (e.g. find alternative data). At the end of this process the findings are presented to the stakeholders and the decision is made whether the data will be used. (Onggo and Hill, 2014, p.199ff)

After collecting the necessary data, they have to be structured. Robinson, 2004 divides data into qualitative and quantitative data. The main focus of many simulation studies is on quantitative data, represented by numbers such as cycle times, arrival pattern or

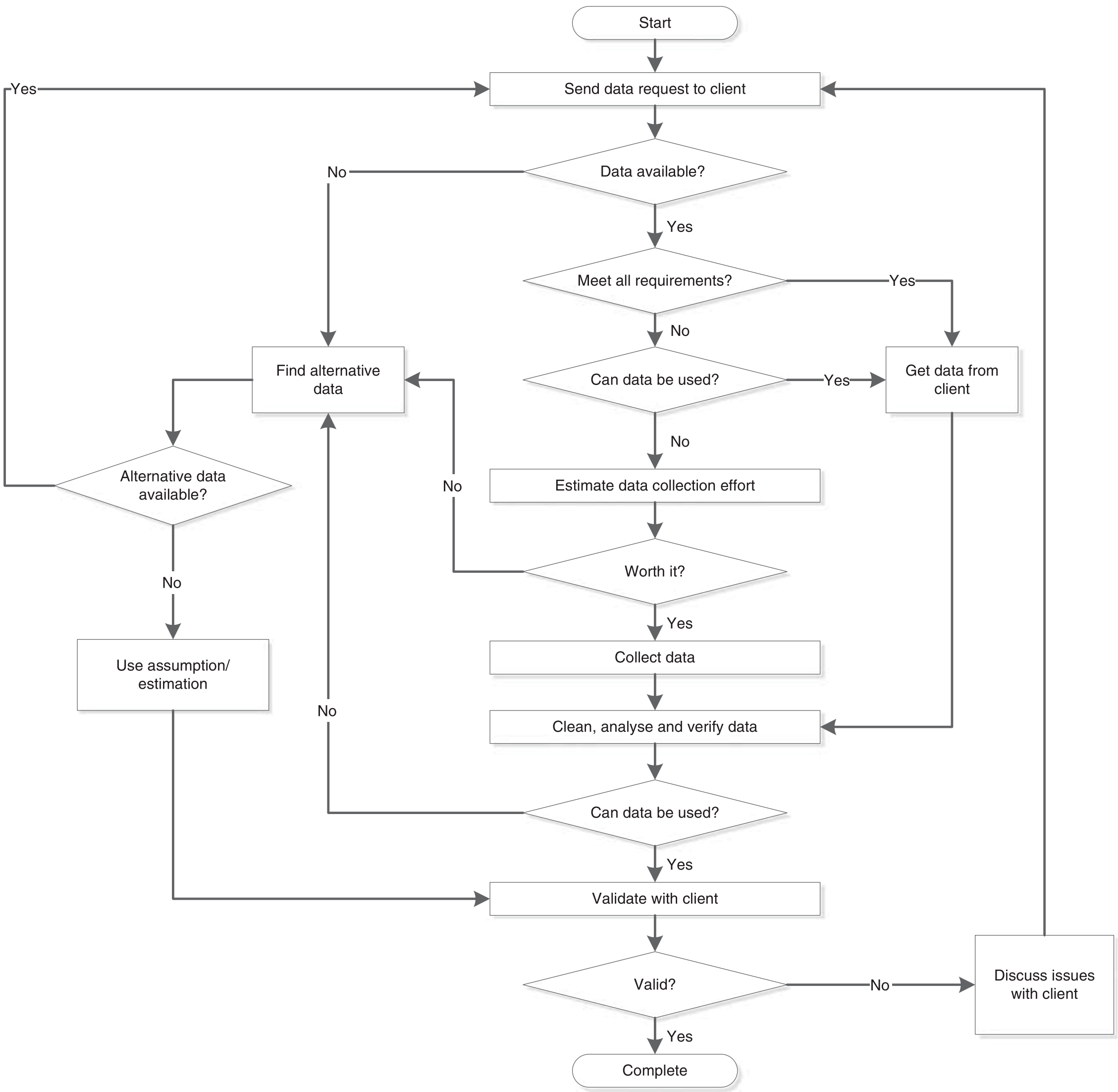


Figure 2.11: Data identification and collection process (Onggo and Hill, 2014, p.200)

breakdown frequencies. However, in a real world system, many facts only can be expressed by non-numerical values like pictures or words. Examples for qualitative data can be CAD drawings of a layout, which defines a manufacturing process. Both, qualitative and quantitative data, have to be considered in the data collection process. In general, gathered knowledge from a system can be also distinguished into information and data. While information can be seen as data with an interpretation, rough data needs to be analysed before being integrated in the model. (Robinson, 2004, p.95ff)

Pidd, 2003 states three groups of data. The first kind is preliminary or contextual data. By asking Why, What, Where, When, How and Who questions about the system, contextual information can be gathered. The objective of this investigation is rather getting a basic understanding about the problem itself then obtaining detailed data. Detailed data is defined by the second group, which is required in order to transfer the conceptual model into a computer model. This (primary quantitative) data describes the components of the model, their properties and how they are connected in detail. Third, data for the model validation is required. In order to avoid using the same data to build the model and validate it, the decision of which data is selected for this stage has to be well thought out. In case of to little data, one possibility could be splitting the data into two sets, one to parametrizes the model and one to check the validity. (Pidd, 2003, p.97ff)

This chapter also concentrates on how to obtain proper data. How can data be collected from the system and furthermore, which data is actually useful for the simulation model and which is not?

The availability of data can be distinguished into three categories (Robinson, 2004, p.97):

- *Category A:* Available data
- *Category B:* Not available but collectable data
- *Category C:* Neither available nor collectable data

Data that falls into *Category A* is data which is available for the modeller. Either they are already known or have been collected previously. Especially modern manufacturing companies have multiple electronic monitor systems which collect data from the operating system. The focus in dealing with this kind of data is rather on ensuring data accuracy and the right format. In most of the cases more data are needed in order to develop the desired simulation. These collectable data are defined as *Category B*. This data collection can be done by means of an electronic data acquisition or by interviewing staff, customers or equipment suppliers. Additionally here, the main priority is on collecting accurate data with the right format. *Category C* defines data, which is needed for the simulation but neither available nor collectable. This can occur when the system has not been implemented yet or is not accessible. It is also possible that no suitable data can be gathered due to a short time horizon of the simulation study. This kind of data, as well as available but inaccurate data, are not uncommon. (Robinson, 2004, p.97ff)

A different approach to catagorize availability of data is provided by Kleijen, 1999. The author distinguishes between three situations:

- no data are available from real situations at all,
- output data are available, but no corresponding information about the input,
- output data and corresponding input data are available.

The latter case with available output and input data is in fact the best and also the easiest one to validate (e.g. regression analysis). The first two situations are faced with a lack of data. The following section discusses different approaches on how to deal with those cases.

2.4.1 Simulation with Lack of Data

The worst case for a simulation study is no availability of data. But even if there is no quantitative information about the system there is still expert knowledge. In order to get necessary quantitative data, a simulation model can be developed. If the input/output behaviour of the simulation does not agree with qualitative data of experts, the model should be questioned and adopted. (Kleijen, 1999, p.647)

In case of a need for expert knowledge Onggo and Hill, 2014 demonstrate in a case study how this situation can be handled. The experts are divided into three groups. Each group represents a certain level of experience. Group one have the least experience (less than two years), the second group are experts with between two and five years of experience and the third with more than five years of experience. This study reveals that junior experts (group one) are more concentrated on requirements of input data and how useful they are and less on the model structure. More experienced experts focus rather on contextual information of the model than on its implementation. These variations of attention among experts needs to be considered in the data collecting process. (Onggo and Hill, 2014)

Beside gathering information from *experts*, Banks et al., 2004 suggest three further ways how to obtain data of a system where no data is available in the first place. In some cases it is possible to collect data from the *nature of the process*. Similar processes often have similar properties. Thus, one way to get data is to look at related systems and adopt their information for the unknown processes. Some real processes also have *conventional or physical limitations*. Those boundaries and limits narrow the input possibilities and must not be ignored. The third way is to look for *Engineering Data*. Companies often provide performance ratings for a process or a product (e.g. failure rates or mean time to failure). This information can be used as starting point for input modelling. (Banks et al., 2004, p.295)

How realistic are simulation studies where almost no data is available? Indeed, there are cases where data are simply not available. Examples for such situations can be found in military case studies like the simulation of a future nuclear war, where almost no historical data is available, fortunately. If a simulation model is built only on assumptions and expert knowledge, finding strong validation claims is impossible. However, the behaviour of the simulation can be investigated by undertaking a what-if or sensitivity analysis. The sensitivity analysis shows how changes of simulation factors influence the behaviour of simulation and respectively its outputs. And if factors with a high significance are changeable and controllable within the real system, the sensitivity analysis provides information how the system can be optimized. Furthermore, it can be used as validation support by showing whether the behaviour agrees with qualitative experts' knowledge. (Kleijen, 1999, p.648)

There are several ways to perform a sensitivity analysis. Many modeller perform a what-if analysis by changing just one factor by time. However, with this method it is not possible to investigate interaction of factors. The Design of Experiments approach (DOE) enables the estimation of two-factor interactions. The core of DOE is defining sets of factor combinations with which the simulation program is executed. The resulting input and output data of these experiments are analysed by use of regression analysis or ANOVA (Analysis of Variance). The importance of individual factors for the simulation depends on their statistical significance. The outcome of the analysis is a regression model (also called metamodel) which describes the behaviour of input and output data. Related to sensitivity analysis is risk analysis. This approach is used in case of not accurately known input factors for the simulation model. Values from respecified probability functions are generated randomly using the Monte Carlo technique. According to the input distribution it is possible to sample all feasible combinations, which further improves the model's credibility. In contrast to that, DOE selects rather extreme combinations with low probability of occurrence in the real-world situation. Compared to sensitive analysis (using DOE) the risk analysis has a much larger number of combinations. (Kleijen, 1999, p.648ff)

Robinson, 2004 describes similar approaches in how to deal with unobtainable data. He defines mainly two ways; estimation of data and treating data as an experimental factor. As basis for estimation decisions several sources can be used. Surrogate data may be obtained from similar systems or organizations. Discussions with staff and equipment

supplier provides good expert knowledge. After all, making just an intelligent guess is still an option.
Needless to say, including estimations in a simulation model reduces its credibility. Therefore, it is very important to document assumptions and simplifications clearly. The author also suggests to perform sensitivity analysis at the experimental stage in order to identify the sensitivity of the results to changes. Especially for sensitive results the aim is to improve estimations. This can be done for example by collecting further data during the life-cycle of the simulation study.
The second approach is treating data as experimental factor. The question is less "What does the data look like?" but rather "How does the data need to look like?". By running different scenarios with varying data, the modeller is able to get a better understanding of which data ranges are feasible. A necessary condition for this approach is that the factors are controllable in the real world situation. If none of both approaches leads to satisfying results it might be better to adopt the modelling objectives or the entire Conceptual Model so the data is not needed anymore. The last option is to abandon the whole simulation study at the cost of the stakeholders who have to make decisions with less or even without any information about the situation. (Robinson, 2004, p.97ff)

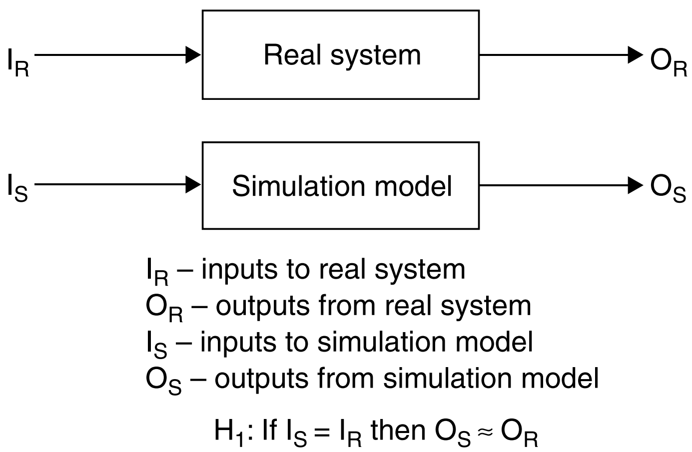


Figure 2.12: Black-Box Validation: Comparison with the real system (Robinson, 2004, p.217)

Figure 2.12 illustrates a validation process for scenarios, as described above. The outputs of the simulation model are compared with the real world outputs. The hypothesis (H_1 : If $I_S = I_R$ then $O_S \approx O_R$) is true, if, with identical input values, the simulation model's outputs are vaguely the same. The author suggests not to compare only average values, but also their spread. Subsequently, a confidence interval can be calculated as follows:

$$\bar{X}_S - \bar{X}_R \pm t_{2n-2,\alpha/2}\sqrt{\frac{S_S^2 + S_R^2}{n}} \tag{2.1}$$

with:

$\bar{X}_S$ = mean of simulated output data

$\bar{X}_R$ = mean of real system output data

$t_{2n-2,\alpha/2}$ = value from Student's t-distribution, with significance level of $\alpha/2$ and $2n-2$ degrees of freedom

S_S^2 = standard deviation of simulated output data

S_R^2 = standard deviation of real system output data

n = number of observations

The number of observations have to be the same for the simulated and the real system. In case of varying sampling size (n) a more complex calculation is needed.

Grey Systems The first scientific paper about Grey System theory was published by Deng, 1982. It refers to systems with a lack of information and was a new approach of dealing with and describing these systems. Different kinds of information qualities are distinguished into White, Grey and Black. *Black* data has a maximum variability (with boundaries $-\infty$ and $+\infty$) and ,thus, is unknown. In contrast to this, *White* data is certain data, which is completely known. All data which is just partly known is denoted as *Grey*. The Grey System theory provides a set of mathematical and statistical approaches, such as grey forecasting, grey decision making or grey generating space, in order to deal with poor data. The objective of this methods is to generate, excavate or extract useful data from incomplete or uncertain data. (Liu and Lin, 2010)

Grey Systems is an extensive field of research. Depending on which type of lack in data, it provides different models, which describe approaches for generating useful data. For further information about this topic, Liu and Lin, 2010 discuss the theory and applications in detail.

2.5 Computer Model and Simulation

Referring to the key stages and processes of Brooks and Robinson, 2000 (figure 2.1) the subsequent step in simulation studies is model coding and building the computer simulation model. Although a simulation can be created with spreadsheets or even with physical models, the focus in this section is on computer modelling. The following sections deals with the key activities of moving from a conceptual model to the computer model.

Simulation modelling systems support the user in the task of model construction, executing simulation experiments and analysis of results. Within each area certain functions need to be fulfilled. The following basic functions can be extended, depending on the requirements of the simulation study (e.g. user interface requirements). (Page, 1991, p.165ff)

Model Construction:

- Input and modification of models
- Storing models
- Access and linking of stored models

Simulation experiment executions:

- Defining of input data for simulation experiments
- Start and execution of simulation runs
- Storing of results

Result analysis

- Access to stored simulation results
- Selection of results for analysing
- Presentation of outcomes

The computer model is a software specific description of the problem. Information and data from the conceptual model are used as basis for the computer model construction. The development of the computer model includes three stages; defining the structure, coding and the documentation of the model. The modeller needs to know the software specifications and properties in order to generate an appropriate model. The first step is to define the structure. It outlines schematically the main components and logic of the simulation plus its variables and attributes. Mainly the nature of the used software defines how this structure is expressed. Complex logics are often difficult to design, especially if the modeller only has very little experience with the software. A remedy for this problem is to start coding a simple simulation parallel. With this prototype, the modeller is able to get a better understanding of the realization possibilities and further is able to try alternative coding approaches. In the model structure designing process four aims should be considered; the speed of coding, code transparency, flexibility and run-speed. (Robinson, 2004, p.128ff)

The subsequent step is developing the computer model itself. This implies three important activities; coding, testing and documenting. Robinson, 2004 suggests to develop the code in small steps, including testing and documenting at each stage. The advantage of this procedure is that errors are identified earlier and the computer model is documented more thoroughly. Another important principle in coding is separation of data. Input data, such as experimental factors and general model data as well as outputs or results have to be separated from the the model code. This makes maintaining and handling of the computer model easer and clearer. (Robinson, 2004, p.129)

The need for documentation in simulation studies is well described in science. However, their importance changed massively the last decades. Whereas Gass, 1984 defines just four main documents (Analyst's Manual, User's Manual, Programmers Manual and Manager's Manual), Robinson, 2004 lists 15 different possibilities of documentation. There are a number of advantages for a well documented simulation model. Large simulation projects often run over a long period of time. A good documentation helps the modeller remember what has been done. Furthermore, in case of personnel change it enables the new operator to understand the current state and the history behind it. A good documentation also support the credibility of the simulation study. It is vitally important for the verification and validation of the model. Most simulation models are not documented well. Oscarsson and Moris, 2002 states that typical costs for documenting are around 20 to 30% of the total costs. But the extent of documentation mainly depends on the complexity of the simulation and on the intended reader.

2.5.1 Simulation Software

The history of simulation software starts in 1955. For the first five years researchers tried to develop reusable routines and search for unifying concepts for simulations. In the early sixties there were just three simulation program languages (FORTAN, ALGOL, and GPSS). A high technological evolution of computer accelerated the development of further program languages. Until 1981 at least 137 simulation programming language were reported. The emergence of high performance (personnel) computers enabled more sophisticated user interfaces and with better graphic representation (2D and 3D graphics). The most recent period is characterized by advancements in web-based simulations a significant higher performance and the combination of simulation and emulation. (Banks et al., 2004, p.86ff)

There are mainly three different possibilities for developing a computer model; programming language, specialist software and spreadsheets.
Using *programming languages* such as Java, C++ and Visual Basic enables the modeller to develop a computer simulation with a high flexibility in model design. Indeed, the process can be time consuming compared to other methods but especially object orientated languages like C++ and Java are very beneficial for building computer simulations. There are a number of *specialist simulation software* available. General purpose simulation packages can be used for a wide range of applications. In contrast to that application orientated simulation packages are designed to focus on one specific field of use such as production scheduling, medical or call centres. In general they are easier to use and the modeller does not need a high level of programming skills, but the features are often limited. For rudimentary simulations *spreadsheet* packages (e.g. Excel) often provide enough functionality. With lookup functions ("VLOOKUP" or "HLOOKUP"), logical functions ("IF", "AND", "OR", "XOR", and "NOT") and random number generator ("RAND") simple time-slice models can be built. Further simulation capabilities can be added with various add-ins or the use of macros or Visual Basic applications. Table 2.1 provides a comparison of different computer simulation approaches as described above. It shows advantages and disadvantages of each approach in a general sense. The selection of the right simulation software mainly depends on the nature of the study (complexity, budget, time frame). (Robinson, 2004, p.41-42)

There is a number of commercial simulation software for discrete event simulation such as AnyLogic, Arena, Enterprise Dynamics, FlexSim, Process Simulater or SIMUL8. However, licences for those applications are very expensive. An alternative to that are open source software like CPN Tools, PowerDEVS or JaamSim. The advantage of those software is a free software licence (GPL - GNU General Public License or Apache 2.0) which allows the user to study, run, share and modify the software for free.

JaamSim: Discrete-Event Simulation Software JaamSim is a Java based, free and open source software for discrete event simulations. It was selected for the case study

Table 2.1: Comparison of program languages, specialist simulation software and spreadsheets (Robinson, 2004, p.42)

Feature	**Spreadsheet**	**Program language**	**Specialist simulation software**
Range of application	Low	High	Medium
Modelling flexibility	Low	High	Medium
Duration of model build	Medium	Long	Short
Ease of use	Medium	Low	High
Ease of model validation	Medium	Low	High
Run speed	Low	High	Medium
Time to obtain software skills	Short (medium for macro use)	Long	Medium
Price	Low	Low	High

of this thesis for several reasons. The licence of JaamSim is Apache 2.0. This licence guarantees the user the freedom to modify, distribute and use the software for any purpose. JaamSim offers a drag & drop user interface (see 2.13) and many possibilities for extensions (e.g. customized object palettes, input and output processing or 3D graphics). A continuous development since 2002 makes this software many times faster than commercial simulation software packages with comparable interfaces. Even large models with up to 40.000 active objects can be run. (JaamSim, 2016)

JaamSim provides a full range of built in objects for developing the computer model (JaamSim, 2016):

- Objects for modelling continuous processes
- Objects for process flow type models, such as queues, servers, etc.
- Text objects for documentation and labelling
- Visualizing objects for outputs
- Graphical objects for logos and background maps

2.6 Application to Construction

This section provides a short overview of peculiarities in the construction industry. One-of-a-kind productions such as construction projects are faced with a high degree of

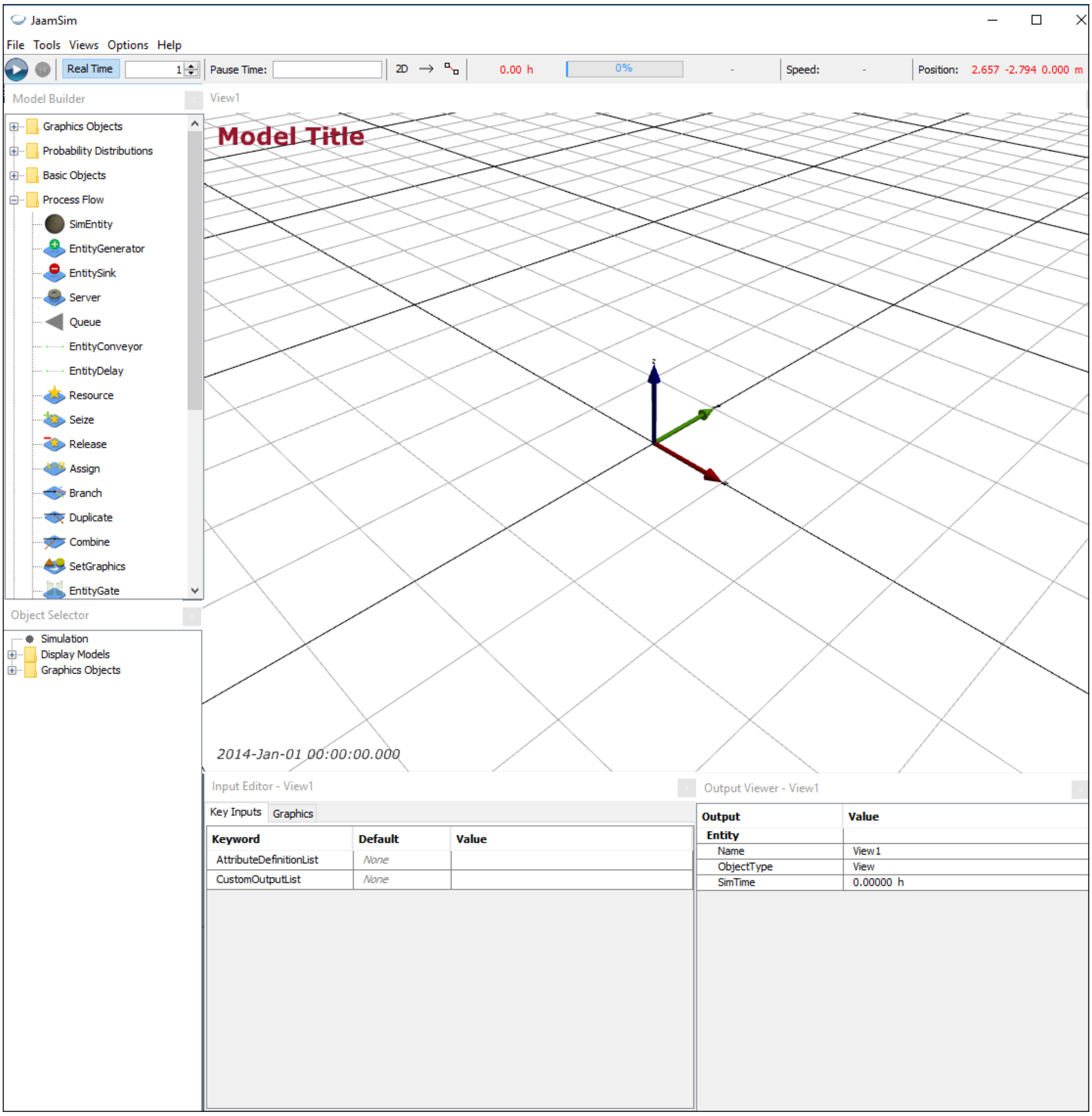


Figure 2.13: Drag and drop user interface of JaamSim

uncertainties. The major challenge is in how to deal with these situations and develop meaningful simulation studies. In particular, the subject of supply chain management is discussed in greater detail.

There are well known chronic problems in construction: insufficient quality, low productivity, poor safety and inferior working conditions. Industrialization, such as modularization and prefabrication approaches, have been viewed as one vision on how to solve these problems in construction. Over decades manufacturing had been seen as a source of innovation and as a reference point for construction. After the manufacturing branch started to study new production philosophies, the construction branch tried to find ways to implement them in their field. Automated and computer integrated construction were possible solutions promoted by researchers. Furthermore, new philosophies such as lean production got established in construction. (Koskela, 1993)

Koskela, 1992 names six types of waste in construction; quality costs, lack of constructibility, poor materials management, excess consumption of materials on site, working time used for non-value adding activities on site, and lack of safety. Each of these types consumes 4-13% of total project costs. The author views the main problems in construction in neglecting the principle of flow. (Koskela, 1993)

The first modern simulation language for application in construction management is CYCLON. Introduced by Halpin, 1977, this model combines all resources, tasks, queues and relationships of construction projects in one model. Over years the model got improved and enhancements were made constantly. The next big step in simulation had been made by introducing object oriented simulation approaches. Well-known modelling and simulation languages in this period were MODSIM (by Shewchuk and Chang, 1991), STROBOSCOPE (by Martinez and Ioannou, 1994) or Symphony (by AbouRizk and Hajjar, 1998). The years since 2000 are characterised by great progress in visual representation of simulation (3D animations) and integration with other tools (e.g. Microstation or AutoCAD). For further discussion about the role of simulation in construction management view AbouRizk, 2010a.

2.6.1 Supply Chain Management in Construction Industry

Supply Chain Management is well described by Schary and Skjott-Larsen, 2001 as follows:

"The objective of supply chain management is to minimize the chain members' total cost of manufacturing, materials, labour, transportation, inventory and information, for all parties concerned. The savings will be shared." (Schary and Skjott-Larsen, 2001)

Supply chain management (SCM) in construction industry initially originates from manufacturing industry. Largely dominated by logistics, SCM today represents a autonomous managerial concept. It offers ideas for reducing problems and waste within the entire supply chain. Concepts from manufacturing such as Just in Time (JIT) or Total Quality Management (TQM) are increasingly being implemented in the construction sector. (Vrijhoef and Koskela, 1999, p.1ff)

In recent years the research effort on construction site simulation has increased massively. The objective of those studies is to develop meaningful simulations in order to identify critical processes. In particular processes of expansion have a high degree of areal and temporal variance. This leads to a system with numbers of different configurations of infrastructure and sub-processes which, in turn, disagrees with the general necessity for simple simulation models. A separation of the system into sub-systems is not appropriate because of their strong interactions between each other. Thus, the main challenge in SCM is determining logistics processes with low impact on the system in order to reduce the modelling and simulation effort. (Voigtmann and Bargstädt, 2010, p.1-2)

Voigtmann and Bargstädt, 2010 describe a general procedure for simulation studies in construction. First of all the entire construction project is analysed and structured into different categories. One example is dividing processes into space forming and not space forming processes. Another good possibility is to separate the system into spatial areas. The subsequent step is to analyse logistics of construction processes. For relevant logistics functions the analyst defines and identifies realization possibilities and organization principles such as storage strategies. Relevant strategies and strategy combinations are gathered and implemented into a simulation model. In the last step, different combinations of construction processes and logistics flows are used for the simulation in order to analyse their influences on the entire system. (Voigtmann and Bargstädt, 2010, p.2)

Figure 2.14 illustrates the different layers of construction supply chain from the source (raw materials/component suppliers, labour market and equipment manufacturer) to the end customer. Of particular importance for this thesis are the materials supply chains.

2.6.2 Pipe Jacking

This section provides the reader with basic knowledge about the tunnelling method which is used at the construction site in the case study.

Pipe-jacking is a trenchless method for installation of underground pipes. Figure 2.15 illustrates a typical pipe jacking arrangement. The underground area consist of three

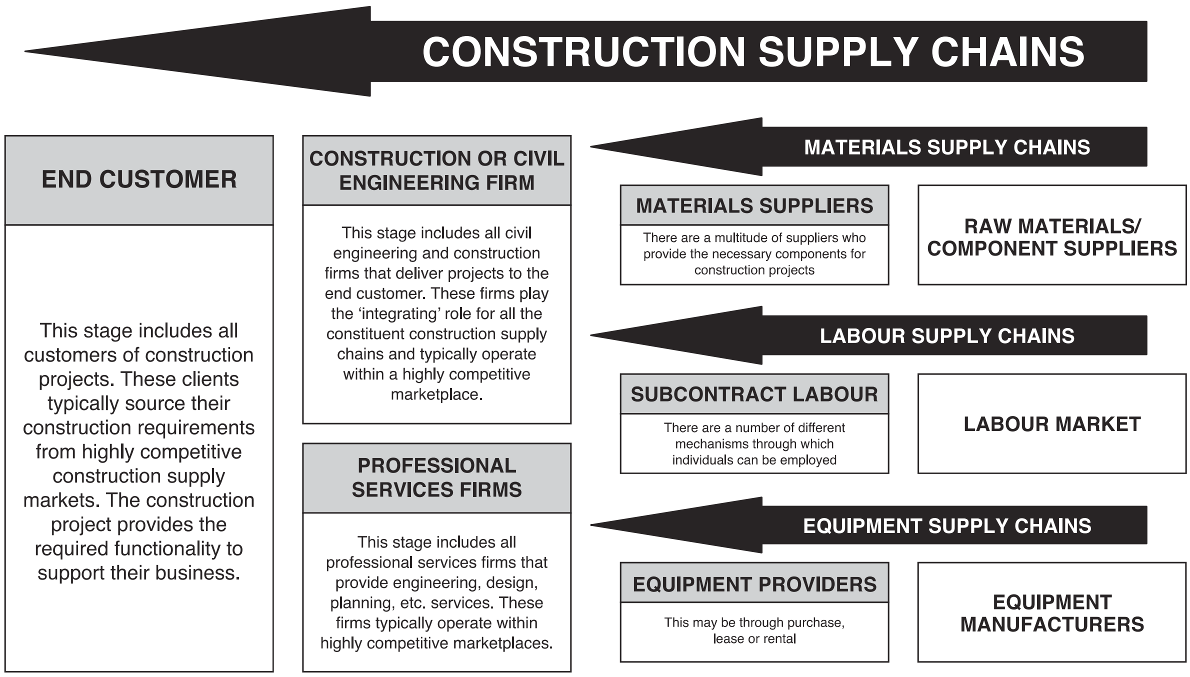


Figure 2.14: The myriad of construction supply chains (Cox, Ireland, and Townsend, 2006, p.411)

main parts; the shaft with hydraulic jacks and the pipe loading area, the inserted pipes with the soil transportation system (e.g. train), the mechanical excavation machine and the tunnel boring machine (TBM) at the front end of the tunnel. The main storage area for pipes and excavated soil is usually located at the surface.

Every pipe jacking cycle starts with the loading process of the pipe. A crane places the pipes from surface down to the underground loading area. There the pipe gets connected with already inserted pipes. Subsequently, hydraulic jacks exert force on the pipe and push the entire tunnel system deeper into ground. The excavation machine is operating during this process and produces soil. After the pipe is fully inserted, the excavated soil is moved through the tunnel to the underground loading area. A crane conveys the tray with soil (muck skip) to the storage area where it gets emptied. This cyclic process is repeated until the desired tunnel length is reached. Afterwards, the tunnel boring machine can be removed from a second shaft at the end of the tunnel.

Pipe jacking has a number advantages compared to the disruptive open-cut construction.

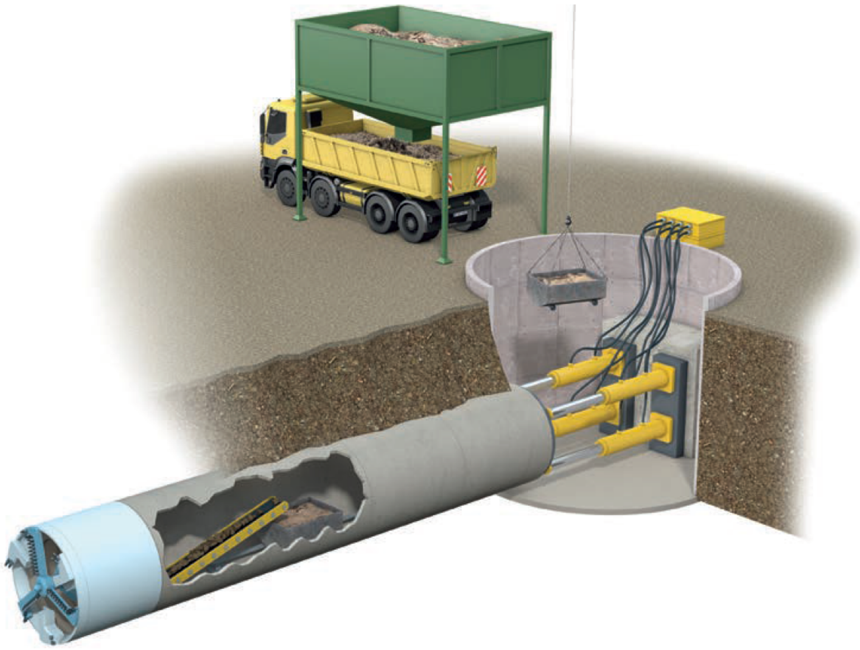


Figure 2.15: Typical pipe jacking arrangement (Pipe-Jacking-Association, 2017, p.4)

Technical benefits are; good flow characteristics due to smooth internal finish, minimal surface disruptions, and low installation time. This is why pipe jacking is especially used in areas with limited space (e.g. cities). Pipe jacking is also an inherently safer method of working and additionally carbon emissions are reduced in excess of 75% compared to open trench constructions. (Pipe-Jacking-Association, 2017, p.4ff)

Chapter 3

Case Study

The task of this case study was to develop an entire simulation study of a construction site's surface operations. A conceptual model had to be created using the *HCCM framework* Furian et al., 2015 and the computer model had to be built on the Java based discrete event simulation software *JaamSim*. The objective was to identify the entire set of supply chain operations and, subsequently, simulate the behaviour of the system. Experimental factors had to be defined in order to get useful findings for future construction projects.

The project team consists, in total, of nine people: three graduate students who were responsible for the project realization, three lecturers from University of Auckland who guided and supported the students with their work and three employees from the construction site's company.

The realization team's responsibilities were structured as follows: the task of the the first Ph.D. student was to build a conceptual model of the entire construction site, the second Ph.D. student was responsible for developing a computer simulation based on this model, and the third graduate student developed an entire simulation study of the materials supply chain. The work of the latter of those three is described in this case study.

At the beginning of the project, the initial real world situation was almost completely unknown to the author. In fact, the following problem description already was part of the conceptual modelling process (Phase 1: Understanding the Problem Situation).

3.1 Problem Description

The New Zealand organization Auckland Transport is improving the public transport situation by building an underground rail line within the central business district of Auckland. As part of this project, storm water pipes have to be replaced. The method which is used to place the pipes underground is called *pipe jacking*. The construction site is located at Victoria Street in the central business district of Auckland. Thus, the site is faced with high frequent and volatile traffic flows. For safety reasons the entire construction site is enclosed with walls and fences. This leads to restricted space for the storage, loading, and waiting area.

The material supply chain problem consists of two main activities, pipe supply and soil removal. Pipe trucks deliver pipes from the supplier to the construction site. Simultaneously, soil needs to be removed. Dump trucks transport soil from the construction site to different disposal sites. Depending on the advancement of the tunnelling process, varying numbers of daily trucks and trips are necessary in order to guarantee a sufficient supply.

Figure 3.1 displays the layout of the construction site. It is structured into four main areas:

- *Pipe Jacking Area,* including the shaft for the pipe jacking process
- *Storage Area,* including the Pipe Storage and the Muck Pit
- *UnLoading Area,* where pipe trucks are unloaded and dump trucks are loaded
- *Waiting Area,* for pipe and dump trucks

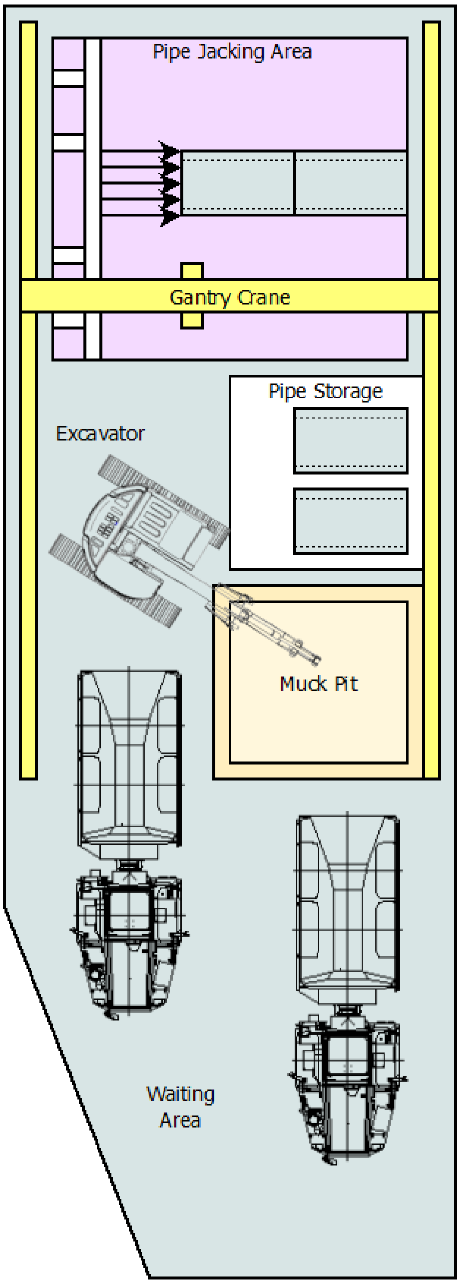


Figure 3.1: Layout of the Construction Site

Chronologically speaking, the construction project is structured into three phases: The preparation phase, the execution phase and the post processing phase. Only the execution phase is of particular interest for this thesis. In this stage the actual tunnelling process starts. In other words, operations before or after this phase are disregarded in this study.

The main objectives of this simulation study can be described as followed:

- Building a complete conceptual simulation model using the HCCM framework
- Developing a computer simulation using *"JaamSim: Discrete-Event Simulation Software"*
- Running simulation scenarios in order to support management decisions for future projects

The following chapters are structured based on these objectives and describe their realization in detail.

3.2 The Conceptual Model

This section documents the process of conceptual modelling based on HCCM framework. It is structured into the four main phases of the HCCM framework. In this case study, the four phase approach is seen as a cycle of continuous development. Thus, the following conceptual modelling process is not structured chronically, but rather content-related.

3.2.1 Phase 1: Understanding the Problem

The objective of this first phase was to fully understand the current situation of the construction site and to discover possible problems. In order to gather proper information, the project realization team held several meetings with responsible persons from three different organizational levels of the construction site; a top-level manager, a middle-level manager, and an assistant for the middle-level management. After several meetings it was noticed, that the perception of the problem varied strongly depending on the organizational level of the person. As one might expect, the top-level manager gave a rather general picture of the construction site's situation and the organizational objectives. On the other hand, the assistant for the middle-level manager provided very detailed information about the sub-processes at the construction site, such as cycle times, breakdown rates, and breakdown durations. Thus, the challenge was to ask the right questions to the right person.

With this in mind, the first aim was to get a basic understanding of the situation and main processes of the construction site's supply chain. A traditional construction supply chain consists of three major parts; Materials, Labour and Equipment. After the first observations, the project team agreed to concentrate entirely on the materials supply chain. At first the construction site by itself was observed and analysed. The questions behind these observations were for instance: How is the construction site structured in general? Which resources are used (human resources, production, inventory, etc.)? What are the main processes?

At this early stage the two main supply chain operations could be identified: *"Pipe Supply"* and *"Soil Removal"*. Another output of these first observations was a rough layout of the construction site and its resources (figure 3.1). After analysing the entities at the construction site, the external supply chain had been observed. Regarding the soil removal there are two dumping grounds, one for deliveries on weekdays and one for weekends. Dump trucks deliver soil from the construction site to the dumping grounds. All main elements of the entire supply chain are listed in table 3.1.

Needless to say, there are many other supply chain operations at the construction site, such as deliveries of technical auxiliaries. However, those operations are generally rare occasions and/or do not have influence on the main supply chain problem. The decisions, which processes are of importance for the simulation are defined later during the model scoping phase.

The following list represents all involving entities, which could be identified after the first observations.

- Truck
  - Dump Truck
  - Pipe Truck
- Construction Site
  - UnLoadingArea
  - PipeJackingArea
  - Gantry Crane
  - Waiting Area
  - Excavator
- Traffic
  - Traffic Supplier

  - Traffic Dumping
- Dumping Ground
- Supplier
- Pipe
- Soil

The next step was to identify the behaviour of the system and its components. Information about individual behaviour had been gathered by interviewing the mid-level manager and his assistant. Table 3.1 lists the defined entities with their associated role within the system. Two main material supply chains could be identified; pipe supply and soil removal.

The HCCM framework suggests different approaches of problem representations. Figure 3.2 shows how the main entities are structured and how they are interconnected.

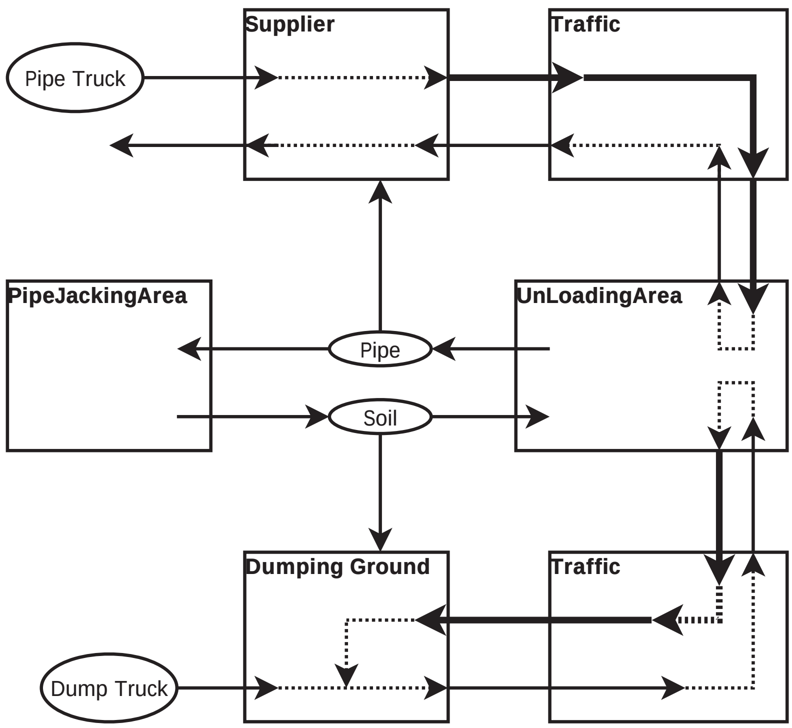


Figure 3.2: Structure of the supply chain (thick line represents loaded conveyances)

General Procedure

The pipe supply chain starts when pipes are requested from the construction site. The

Table 3.1: Entities within the Simulation Model

Entities	
Components	**Explanation**
Truck	Entity container for Soil/Pipes
- Dump Truck	Entity container for Soil, delivers Soil to Supplier
- Pipe Truck	Entity container for Pipes, delivers Pipes to Construction Site
Construction Site	Entire area of construction site
- UnLoadingArea	Storage area where DumpTruck is getting loaded and PipeTruck is getting unloaded
- PipeJackingArea	Represents the tunnelling operation, consumes Pipes and produces Soil
- Gantry Crane	Unloads Pipes from PipeTruck, moves Pipes from UnLoadingArea to PipeJackingArea, and moves the muck skip from PipeJackingArea to UnLoadingArea and back
- Waiting Area	Area, where Trucks are waiting for getting un/loaded
- Excavator	Entity, which loads the DumpTruck with Soil
Traffic	General area, where DumpTrucks and PipeTrucks are going from or to the Construction Site
- Traffic Supplier	Traffic from and to Supplier
- Traffic Dumping	Traffic from and to Dumping Ground
Dumping Ground	Area, where Dump Truck is getting unloaded
Supplier	Area, where Pipe Truck is getting loaded with pipes
Pipe	Flow entity, issued by Supplier, consumed by PipeJackingArea
Soil	Flow entity, issued by PipeJackingArea, consumed by Dumping Ground

pipe truck is (always) loaded with two pipes at the pipe supplier site. Subsequently, the truck travels from the supplier to the construction site, where it is unloaded by the gantry crane. After the unloading process the pipe truck travels back to the supplier. The pipes can be stored in two areas, one of which is located at the surface and the other down in the shaft. In case of an emergency, additional space for two pipes is available near the actual construction site. In total the pipe storage capacity is 10 (+2 external).

The pipes in the pipe storage are then waiting to get loaded into the pipe jacking area by the gentry crane. After finishing the pipe loading, the pipes are prepared for injection.

Subsequently, the actual pipe jacking starts. The entire pipe system is pushed by powerful jacks towards tunnelling direction. At the end of the tunnel, a tunnel boring machine (TBM) excavates soil and places it into a muck tray located behind the TBM. As soon it is filled to maximum capacity, the muck tray travels back to the shaft, where it is picked up by the gantry crane and emptied at the soil storage. The pipe jacking and excavation process has to be stopped until the muck tray is back at the TBM.

Each day a certain number of dump trucks are ordered. As soon a dump truck arrives at the site, the excavator loads the truck with soil. After finishing the loading process, the truck travels to the dump ground where it is unloaded. Depending on the weekday there are two possible dump grounds, one for deliveries during the week and one for weekends. The soil removal procedure is a cyclic and repetitive process. It is stopped as soon the muck pit at the construction site is emptied.

More detailed information, such as cycle times, travel times, or capacities, had been gathered after defining the model's scope and level of detail (section 3.2.4).

3.2.2 Phase 2: Identification of Modelling and General Objectives

The process of identifying modelling and general objectives actually happened parallel to the first phase. At periodic meetings the situation had been discussed with the responsible persons from the construction site. The first and most important organizational aim is to guarantee that there are *no pipe jacking delays caused by surface problems.* Interruptions of the pipe jacking process are very cost-intensive and have to be avoided under all circumstances.

There are three cases, in case of which the pipe jacking process needs to be stopped.

- Breakdown of the pipe jacking machine or supporting devices
- Maximum capacity of the muck pit is reached
- Pipe storage level is zero

Breakdowns cause an abrupt stop of the tunnelling process. Unexpected incidents usually occur due to electrical faults, mechanical faults, problems with the ventilation or water system, and problems with the conveyor. Breakdowns occur rather independent from supply chain processes. Thus, the focus in this study is on the muck pit level and the pipe storage. But breakdown rates are not ignored. The construction site's productivity

Table 3.2: Simulation Study Objectives

Organizational Aim	
No interruption of the pipe jacking process due to problems on the surface (transport or storage)	
General Objectives	
Project duration	4 months
Workload	1 Developer
Visualization requirements	Visualization of the computer model with JaamSim
Reusability requirements	The model should be designed with respect to changing conditions for future transport problems.
Documentation and usability	The simulation should be easy to handle and easy to understand, even for untrained personnel. A detailed documentation is required.
Modelling Objectives	
Policy improvements	Evaluating different policies for truck utilisation including ordering.

is characterized by a great number of small and large breakdowns. This data needs to be analysed and is vitally important for the subsequent computer simulation.

With this in mind, two sub-objectives could be identified. First, the muck pit level must not reach the maximum and second, the pipe storage level must not reach zero at any time. In order to guarantee a sufficient and efficient supply, the modelling objective is to define, simulate and evaluate different ordering policies.

The HCCM framework suggests to define the simulation study objectives in tabular presentation. Table 3.2 shows the definition within the conceptual model.

As it is often the case in conceptual modelling, the definition of the objectives had been changed over time. One result of a first simple simulation was, that there were almost no cases where the muck pit reached the maximum level, respectively the pipe storage capacity had reached zero. Hence, the focus was rather on finding efficient order policies in order to avoid wasting time, such as idle truck waiting time at the construction site.

Another initial objective was to reduce pipe storage costs at the supplier. The construction company has to pay additional fees to the supplier for extra pipe storage time in case of a project delay. After some investigation, the project team concluded that there were no ways to reduce this time by improving the supply chain, because the greatest part of delays was caused by breakdowns.

3.2.3 Phase 3: Defining Inputs and Output Responses

Based on the objectives, the following phase was to define input and experimental factors as well as output responses of the simulation.

Experimental Factors

Experimental factors are used to simulate different scenarios. By varying those inputs, the behaviour of the simulation can be examined. The prerequisite of defining those variables is a modifiability within the real situation.

Capacity Muck Pile: The initial muck pile capacity of 100 m^2 did not cause any problems yet. A smaller muck pit area could improve the space conditions at the construction site. The question to answer was; What is the minimum necessary capacity of the muck pit?

Target Value Pipes: The target value for the pipe storage defines at which threshold new pipes were to be ordered. The advantage of a low average pipe storage would be a improvement of the the space requirements.

Number of pipe and dump trucks: One of the aims is to find the best configuration of available trucks. The optimal solution is by meeting the objectives with as little aggregated truck waiting time as possible.

Travel times to supplier and dump ground: The impact of alternative dumping grounds and supplier is simulated by varying travel times. Data is collected by means of *Google Maps Directions API*.

Table 3.3 shows the experimental factors in tabular form as listed in the conceptual model.

Table 3.3: Experimental Factors

Experimental Factors	
Capacity muck pile	Capacity of the muck pile at the construction site
Target value pipes	Pipe storage target value at the construction site
# Pipe trucks	Number of trucks available for pipe transport
# Dump trucks	Number of trucks available for transport soil to dumping ground
Travel time - supplier	Travel time distribution from and to supplier - Google Directions data
Travel time - dumping ground	Travel time distribution from and to dumping ground - Google Directions Data

Input Factors

Input factors are constant values which represent key characteristics of the system. Those numbers can be adopted for future projects by changing the computer simulation. Input factors are:

- Total number of pipes
- Soil issued per pipe
- Breakdown times and types
- Operating hours of trucks and construction site
- Duration for loading and unloading soil at the dump ground and construction site
- Duration for loading and unloading pipes at the supplier and construction site

Simulation Outputs

The output responses of the simulation have to be defined according to the simulation objectives and aims. The following outputs were determined:

- *Mean waiting time for trucks:* The average time a truck has to wait to get loaded or unloaded at the construction site
- *Total time of delay:* The total project delay, caused by surface (supply chain) problems
- *Total simulation time:* Represents the total project time

- *Maximum soil level:* The maximum level of the muck pit

The final definitions of experimental factors, inputs and outputs in tabular form can be found in the Conceptional Model in the Appendix.

3.2.4 Phase 4: Defining the Model Content (Scope/Level of Detail)

The subsequent step is to specify content of the model. In the first step, the model had a very high level of detail. After the data acquisition, attempts were made to include all activities and entities of the system. It could be seen from the first computer simulations, that a too detailed model had more disadvantages than advantages. Thus, the aim was to build the simulation only as detailed as necessary, but also as simple as possible. This lead to some simplifications, which are documented in this section.

Model Structure

In order to get a rather simple and understandable simulation model, simplifications had to be made. Every entity from the data acquisition was analysed and a judgement was made whether to include it in the model or not. Some entities were represented by other parent entities (e.g. Gantry Crane and UnLoadingArea) and, hence, are excluded. Others are represented only as attributes of other entities (e.g. Pipe and Soil). Table 3.2 lists the entity structure with a justification for each of them.

The structure of activities and processes in the system are described in the following section.

Simplifications & Assumptions Due to the nature of any real world problem, it is not possible to gather all information and data of a system. Table 3.5 lists main simplifications and assumptions, made in the conceptual model. Their confidence and the influence on the simulation is assessed, based on the personnel view of the modeller. For instance, breakdown rates define to a considerable extend the total project duration and, thus, have a high impact to the simulation model. On the other hand, it is obvious that trucks need to be refuelled but this, rather short and rare event, does not influence the model much (low impact).

Table 3.4: Entities within the Simulation Model

Entities		
Component	**In/Exclude**	**Justification**
Truck	Include	Entity container - required flow entity
- Dump Truck	Include	Entity container for Soil - required flow sub-entity
- Pipe Truck	Include	Entity container for Pipes - required flow sub-entity
UnLoadingArea	Include	Contains all loading and unloading activities as well as access polices
PipeJackingArea	Include	Tunnelling operation, consumes Pipes and produces Soil
Traffic	Include	Contains traffic data - modelled as delay for Trucks
- Traffic Supplier	Exclude	Traffic from and to Supplier of Pipes
- Traffic Dumping	Exclude	Traffic from and to Dumping Ground
Dumping Ground	Exclude	Is represented by the Soil Unloading activity
Supplier	Exclude	Is represented by the Pipe Loading activity
Gantry Crane	Exclude	UnLoading Area represents behaviour of Gantry Crane
Excavator	Exclude	UnLoading Area represents behaviour of Excavator
Pipe	Exclude	Handled as attribute of other entities
Soil	Exclude	Handled as attribute of other entities
Construction Site	Exclude	Represented by UnLoadingArea and PipeJackingArea
Waiting Area	Exclude	Represented by UnLoadingArea

Model Individual Behaviour

In *Phase 1: Understanding the Problem* (section 3.2.1) a first overview of the problem could already be generated. After defining the scope and structure of the model, the individual behaviours of the entities needed to be gathered. In meetings with the mid-level manager and his assistant more detailed information about the processes could be collected. The main objective in this section was to form a broad description of all

Table 3.5: Assumptions and Simplifications within the model

Assumptions	**Confidence**	**Impact**
Same breakdown rates for future projects	Medium	High
Simplifications		
No severe traffic breakdowns in model considered	Low	Low
PipeJacking and UnLoading activities do not influence each other directly (exclude gantry crane)	Low	Medium
Exclude day 1 in Simulation (day 1 is represented by soil/pipe start value)	Medium	Low
All trucks are from the same operator (actually 1 external and 1 internal)	Medium	Low
No "Waiting for Unloading" at DumpingGround	High	Low
No delay at Supplier caused by loading queue	Medium	Low
No fuel refill necessary	Low	Low
Pipe storage down the shaft not considered	Medium	Medium

activities at first in textual form and later with visual presentations. The two main parts of the supply chain are pipe supply and soil removal.

Pipe Supply

One pipe truck (capacity 2 pipes) delivers pipes from the supplier to the construction site. The quantity of delivered pipes depends on the advancement of the pipe jacking process and varies between zero and six pipes, respectively zero and three trips per day. The main sequence of the pipe supply can be described as follows. If pipes are requested, the truck is loaded at the pipe supplier. After the loading process the truck goes to the construction site. This travel duration varies depending on the traffic situation. After arriving at the construction site, the pipes are unloaded by the gantry crane. The crane is not available at all times, thus the truck has to wait until the gantry crane is available. After unloading the pipes, the truck goes back to the supplier. Ideally, this procedure is repeated up to three times a day, at 10:00 am, 1:00 pm and at 16:00 pm. However, deliveries can be cancelled due to changing demand caused by delays in the pipe jacking process.

Soil Removal

Depending on the advancement of the pipe jacking process, different amounts of soil are produced and must be stored in the muck pit. When an emptying of the muck pit is necessary, one or more dump trucks are ordered. After arriving at the construction site an excavator loads the truck. The fully loaded truck transports the soil to the dumping ground. This travel duration varies depending on the traffic situation. This procedure is repeated during the operating hours for each truck until the muck pit is empty. Further constraints define the loading and unloading process at the construction site; only one truck can be served (loading soil or unloading pipes) at a time. There is room for a maximum of one additional truck, waiting to be served. The process of unloading pipes has the highest priority, however, the loading process of a dump truck cannot be interrupted.

In order to provide the simulation programmer with a sufficient insight into the real world situation, several approaches of representing the behaviours were undertaken. Figure 3.3 displays a first overview of entities and their behaviour. The entities are distinguished with colours; PipeTruck - red, UnLoadingArea - white, PipeJacking - blue, and DumpTruck - beige. Additional to the predefined entities, pipe and soil flow is added in order to illustrate their flow. Furthermore, the figure shows the predefined activities of each entity. A first description of the activities is listed in table 3.6.

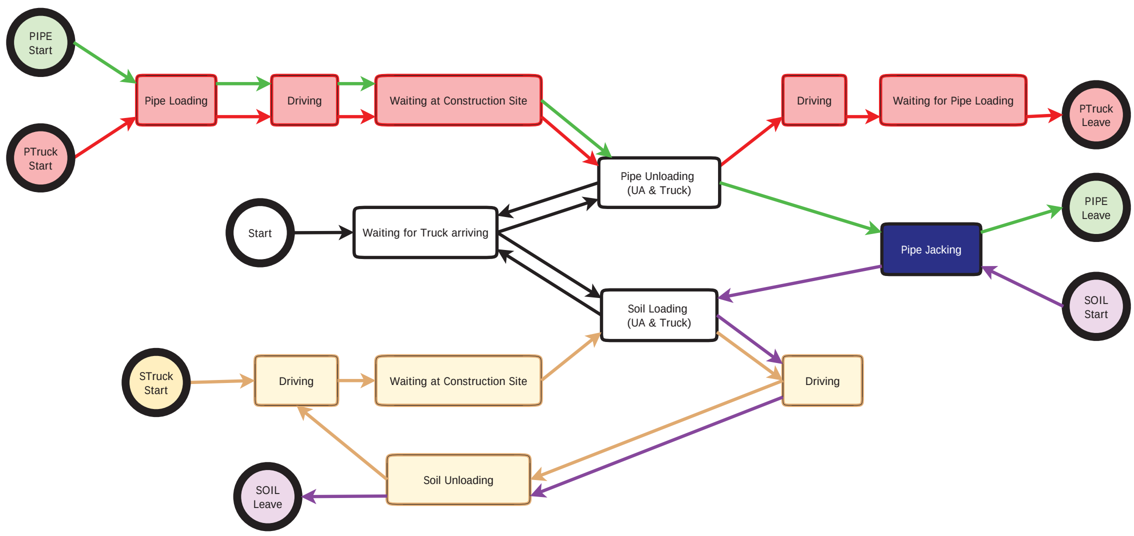


Figure 3.3: Overview of entity flow in System, supplemented with pipes and soil

Based on the entity flow overview (figure 3.3) the behaviour of individual entities was

Table 3.6: Activities within the Simulation Model

Activities		
Component	**In/Exclude**	**Description**
Waiting at Construction Site	Include	Activity for PipeTruck and DumpTruck
Soil Loading (UA)	Include	Loading soil from muck pit onto DumpTruck at the UnLoadingArea
Soil Loading (Truck)	Include	Activity of DumpTruck during Soil Loading (UA)
Pipe Unloading (UA)	Include	Unloading PipeTrucks at UnLoadingArea
Pipe Unloading (Truck)	Include	Activity of the PipeTruck during Pipe Loading (UA)
Driving	Include	The activity of Trucks being in Traffic
Pipe Loading	Include	Loading pipes onto Pipe Truck
Soil Unloading	Include	Unloading of DumpTruck at the deposit ground
Waiting for Pipe Loading	Include	PipeTruck waiting for pipes getting loaded
Pipe Jacking	Include	Includes all activities from pipe consumption (input) to soil production (output)
Gantry Crane	Exclude	Unloads PipeTruck and transports Soil from pipe jacking area to muck pit (out of scope)

extracted. One example, how this was realized, is the behaviour of the dump truck as shown in figure 3.4. The illustrations of the residual entities can be found in the CM in the Appendix.

Another approach to show the individual behaviour is to describe activities in tabular form with detailed information about certain conditions and states. One example of an activity description is shown in table 3.7. It provides information of start and end types (Requested, Sequential, or Scheduled), changes of state, and attributes of the activity. The complete detailed list of all activities can be found in the Appendix.

All of the information provided up to this point already gives a good picture of the real world system and how it can be implemented into the simulation. However, one aim was

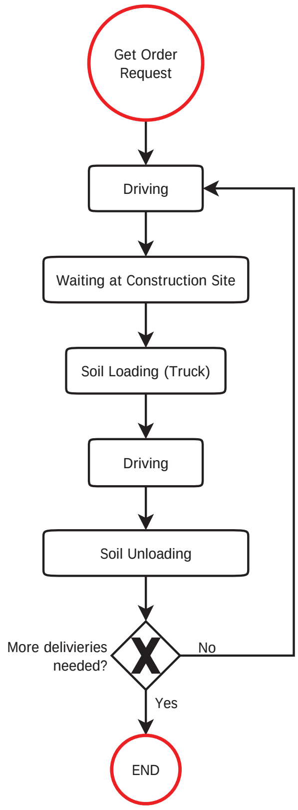


Figure 3.4: Behaviour of the DumpTruck

to find one type of presentation with which it was possible to show the entire system with all entities, activities and their connections. The *Business Process Model and Notation* (BPMN) enables such a visualization. The traditional BPMN does not have particular notations for control units (paragraph *Model Control (System Behaviour)*, page 59). An attempt has been made to include control parts of the HCCM into this approach. Figure 3.5 shows all main activities and events and their connections.

Table 3.7: "Soil Loading (UA)" Activity

Soil Loading (UA)	
Participating Entities	UnLoadingArea, DumpTruck
Start Type	Requested
End Type	Scheduled
Start State Changes	none

End State Changes	UA.SoilLevel -= DumpTruck.Capacity
	# UA.Trucks -= 1
	DumpTruck.RouteToDumpingGround = TRUE

Attributes	**Description/Value**
Duration	Input data [min]
DumpTruck.Capacity	Input data [m^3]
Request Attributes	**Description/Value**
TimeRequest	When was the request made from Truck Access Control Unit
Request Specification	Request filed after Soil Loading

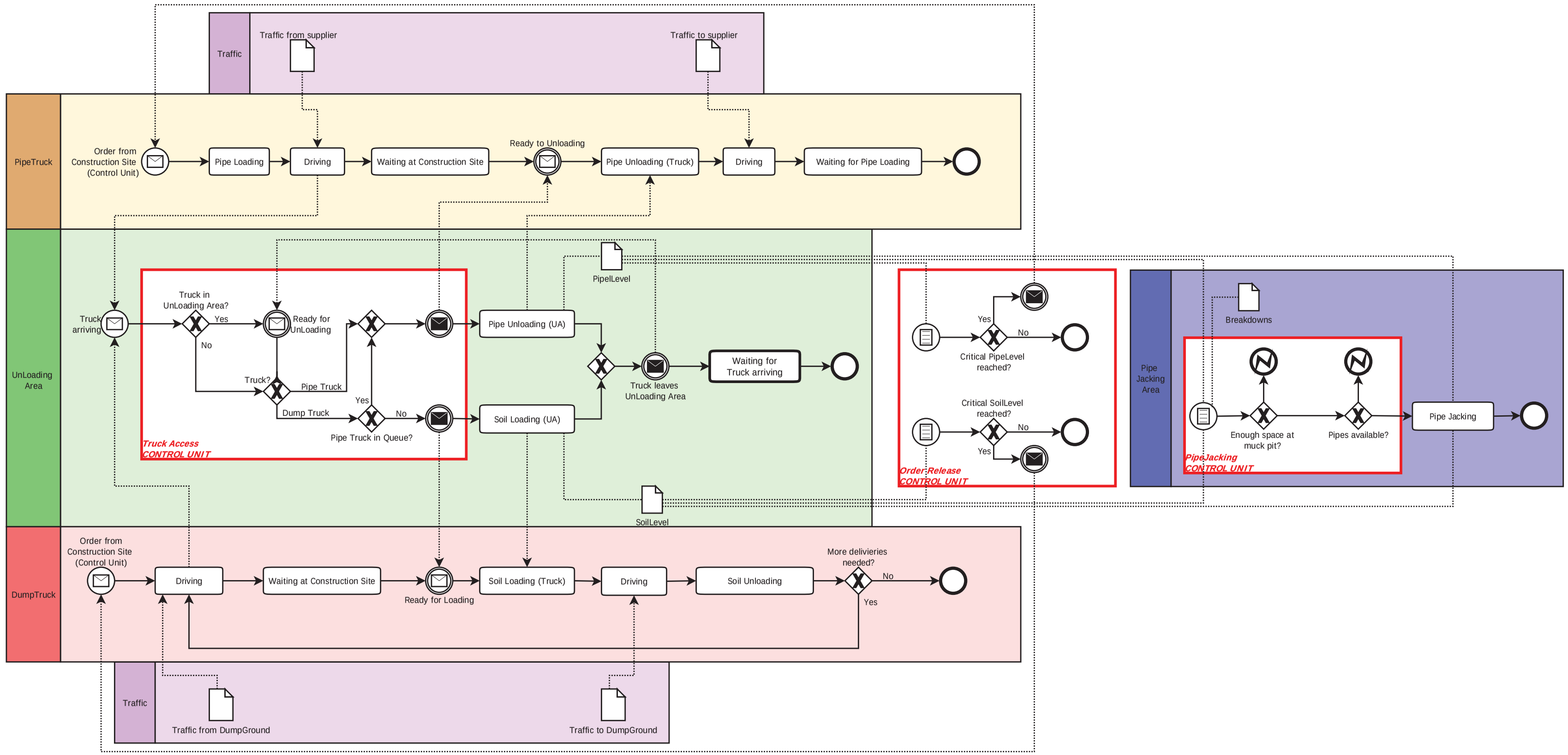


Figure 3.5: Behaviour of UnLoadingArea entity

Model Control (System Behaviour)

The model control is a main feature of the HCCM framework. Control units can be a set of strategies or rules. Their task is to control the behaviour of the defined entities. In the real world situation three areas were identified, at which control mechanisms were required; PipeJacking Control Unit, Truck Access Control Unit, and Truck Order Control Unit.

The general structure of the model control is shown in figure 3.6. Octagons in this figure represent the control units, which control the activities, displayed with rectangles. The PipeJacking control unit handles the Pipe Jacking activity and decides whether this process has to stop. This case occurs if there are problems with the supply chain (pipe delivery and soil removal). The Truck Access control unit regulates the access of trucks into the construction site and the third control unit manages the ordering process of pipe and dump trucks.

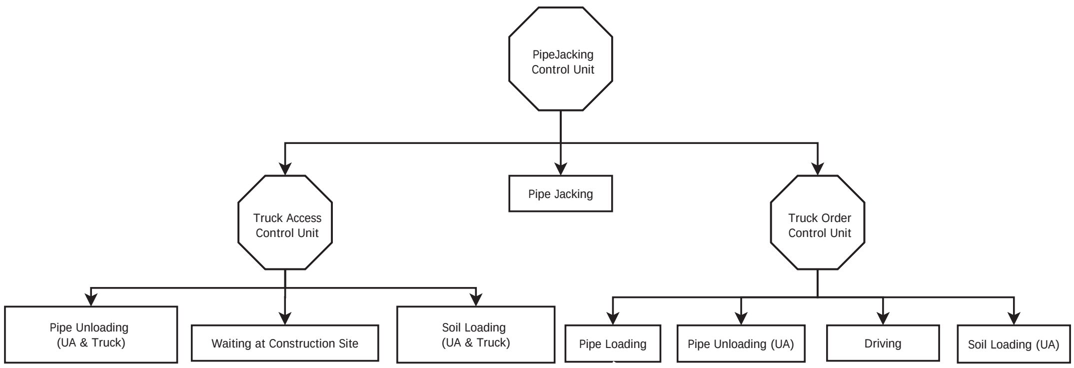


Figure 3.6: Control view of the problem

Additionally to the structural view, each control unit is described by a set of rules. For instance, at the construction site there is a specifically defined policy, how trucks are granted access. Pipe deliveries have priority one, this means other deliveries have to wait until the pipe unloading activity is finished. One exception is the case in which the capacity of the muck pit has reached its maximum. In order to avoid delays of the pipe jacking process, the soil needs to be removed immediately. Figure 3.7 shows the visual representation of this policy.

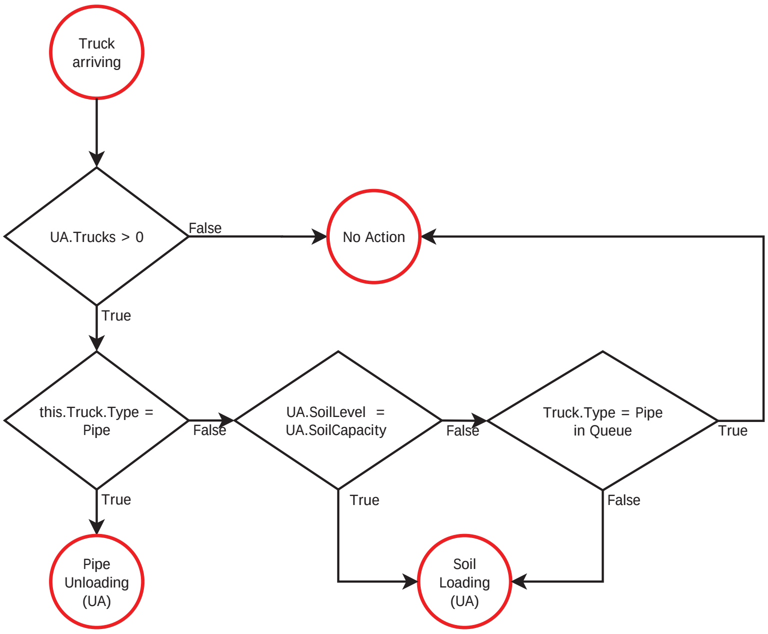


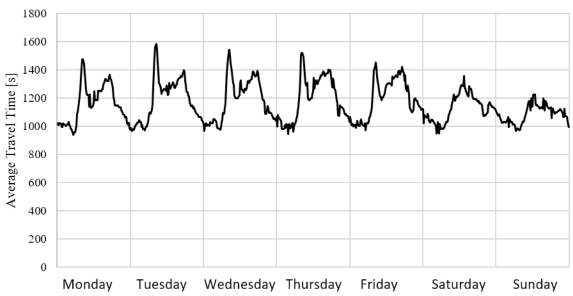
Figure 3.7: TruckAccess Control Unit rules

3.2.5 Data Acquisition and Validation

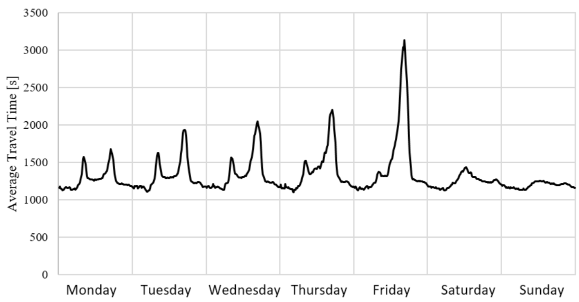
One of the biggest challenges of this case study was not only gathering information and data from the system, but rather to interpret and integrate it into the computer simulation. Unfortunately, some data was not available, because some processes had not been investigated yet by the company. Nevertheless, in order to get intelligence, the mid-level manager and his assistant were asked for their expert knowledge and insight. One example for such a case, was the variance of truck travel durations. After consultation with the responsible persons from the construction site, a maximum and minimum travel time could be defined. The aim was to find more meaningful and accurate data, especially for critical processes. This section describes the process of data acquisition and the analysis of truck travel times and breakdown times.

Travel times

Gathering information on the truck travel times was one challenging part of the case study. Due to a heavy city traffic around the construction site, the trucks are faced with a highly volatile traffic situations. The volume of traffic varies strongly depending on the weekday and daytime. After the first round of interviews with the construction site's managers, it became clear that the needed information was not available in that form, although it was required for the simulation. The objective in this part was, to find sufficient data about travel times. One possibility would have been to track the trucks via GPS (Global Positioning System). This idea was discarded soon, because the construction project was already in its final stage. It would have taken too long to set up the technical equipment and just very little data could have been gathered within this short period of time.



(a) Dump Ground to Construction Site



(b) Construction Site to Pipe Supplier

Figure 3.8: Travel time predictions with traffic (© 2015 Google Inc, used with permission. Google and the Google logo are registered trademarks of Google Inc.)

Another idea was to collect data from internet databases and maps. For this kind of information, there is a number of different providers such as Google Maps, Microsoft Bing Maps, OpenLayers or Foursquare, to name but a few examples. All of them provide access to the data through an Application Programming Interface (API). With help from the Faculty of Engineering (University of Auckland) a program was developed, which collects data from Google Maps Directions API. With this tool it was possible to gather quarter hourly predictive travel times, based on historical data. Figure 3.8 shows two examples of datasets for travel times.

Validation

For the purpose of validation, this data was compared with information, gathered from interviews. The average travel time for dump trucks in the simulation varies between 0.8 and 1.2 hours. The data analysis of the real system show variances between 0.3 and 1.5 hours (figure 3.9). After consultation with the project team it could be agreed that travel times below 0.5 hours are unfeasible. Furthermore, travel durations over 1.5 hours only can occur during severe traffic breakdowns or during idle times of the driver. Both cases are excluded in this simulation study (see assumptions and simplifications - table

3.5). The average travel times of the simulation reflect the real travel time situations and thus, can be considered valid and suitable for the simulation.

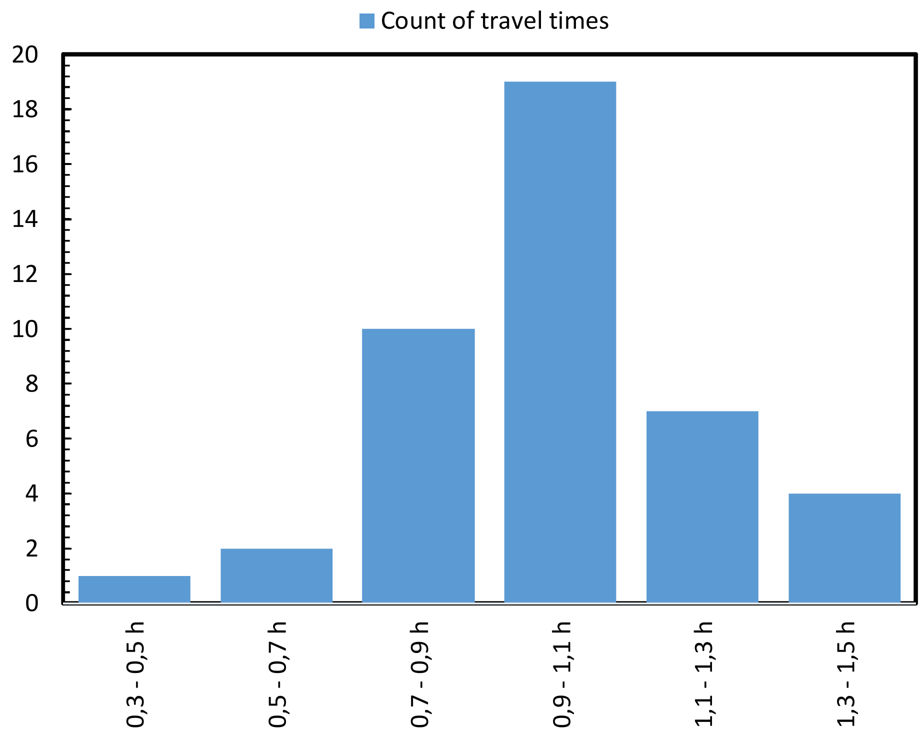


Figure 3.9: Count of duration between departure and arrival of dump trucks

Breakdowns

Another big part in the data acquisition was the integration of breakdown rates in the model. The breakdown rate was given as hours per 12-hour shift for each component at the site. With this kind of data it was not possible to determine when and how long certain breakdowns occurred (e.g. one electrical breakdown over two shifts was displayed as two breakdowns). Figure 3.10 shows a schematic example of breakdown data on the construction site. It is structured into six main breakdowns; Mechanical Fault (M), Electrical Fault (E), Water System (W), Surveyor (S), Ventilation System (V) and Others (O). Additional to this data set, net working times of the pipe jacking process were given for each shift. The challenge was to find a way to integrate this kind of data into the simulation. In order to solve this problem, several approaches had to be tested. The first idea was to just look at the net pipe jacking time and vary the breakdown durations and intervals until the computer simulation net pipe jacking time corresponded with the real time. The simulated breakdown rates however, were unrealistic compared to the real data.

	Mechanical Fault	Electrical Fault	Water System	Surveyor	Ventilation System	Other	SUM
Shift 1	1	0,75					1,75
Shift 2	1,5			2	5		8,5
Shift 3			12				12
Shift 4			12				12
Shift 5	0,75	2,25	0,5				3,5
Shift 6				1		11	12

Figure 3.10: Schematic representation of collected breakdown data

Another idea was to look at every individual type of breakdown and generate breakdown rates with *Markov Chains*. The prerequisite for using the markov chain approach is that the occurrence of every event (breakdown) only depends on the previous event. The first step was to analyse the different breakdown combinations and their occurrence during each shift. 15 different breakdown cases were defined. Table 3.11 shows the breakdown makrov chain in tabular form. The values in the table describe the probability with which a certain breakdown state is followed by the previous one. Breakdown states are any combinations of breakdowns including no breakdown (none). As an example, one shift with a mechanical and electrical breakdown (row - EM) is followed by a shift with a mechanical breakdown (column - M) with a probability of 30%.

The results of this method were not usable for two reasons: It could not be proven that the occurrence of a certain breakdown state only depended on the previous state (prerequisite for *Markov Chains*) and the second problem was that JaamSim does not provide the environment for *Markov Chain* implementation.

The challenge was to find a way to integrate the breakdown data into the simulation software. The final approach was to split up the breakdown data into two sets; long breakdowns ($>$ 12 hours) and short breakdowns ($<$ 12 hours). The resampling approach, used for the simulation, is called *bootstrapping*. That means, that new breakdown scenarios were generated for each simulation run, based on the real world data. A schematic representation of input data for short breakdowns is shown in figure 3.12.

Validation

The second parameter for breakdown definition in JaamSim is breakdown interval. This value was handled as an experimental factor. The breakdown interval was defined as a gamma distributed value between 0 and 12 hours.

	E	EM	EMS	EMSO	EMV	ES	ESO	M	MS	MSO	MVS	S	W	MW	none
E	40%	20%						10%	10%				10%	10%	
EM	9%	9%	9%		9%	9%	9%	36%	9%						
EMS	33%	33%		33%											
EMSO			100%												
EMV	33%	33%	33%												
ES		100%													
ESO										100%					
M		10%			10%			40%			10%				30%
MS	33%	67%													
MSO												100%			
MVS		100%													
S															100%
W															100%
MW					100%										
none	11%	11%						11%	11%						56%

Figure 3.11: Makrov table of breakdown combinations

	Mechanical Fault	**Electrical Fault**	**Watering Systems**	**Surveyor**	**Ventiliation System**	**Other**
Probability	34%	14%	18%	19%	11%	4%
Distributed	exponential	exponential	exponential	exponential	exponential	exponential
Mean duration [h]	2,39	2,20	2,88	0,78	1,64	4,81

Figure 3.12: Schematic representation of short breakdown duration input

According to the Black-Box validation in section 2.4.1, the shape factor and mean time (= input factor I_S) were altered until the total simulation time equalled the actual total project time ($O_S \sim O_R$, with $O_R = 62$ days and $O_S = 62,419$ days). After analysing the results, it could be concluded that this method delivered the most meaningful results in this case.

3.3 Computer Simulation

The subsequent step in the simulation study was to develop a computer simulation, based on the conceptual model. As already mentioned, the final solution as presented in the case study is a result of many alterations and iteration.

The first version of the simulation was a very basic one, with the soul purpose of getting an idea of how the system could work. But after further interviews, detailed information about the procedures could be gathered and the simulation became very complex. It soon became clear that a complicated computer simulation would not lead to meaningful results. Some necessary information could not be gathered from the construction site and therefore assumptions and simplification had to be applied (compare section 3.2.4). Figure 3.13 shows the front-end of the final computer model.

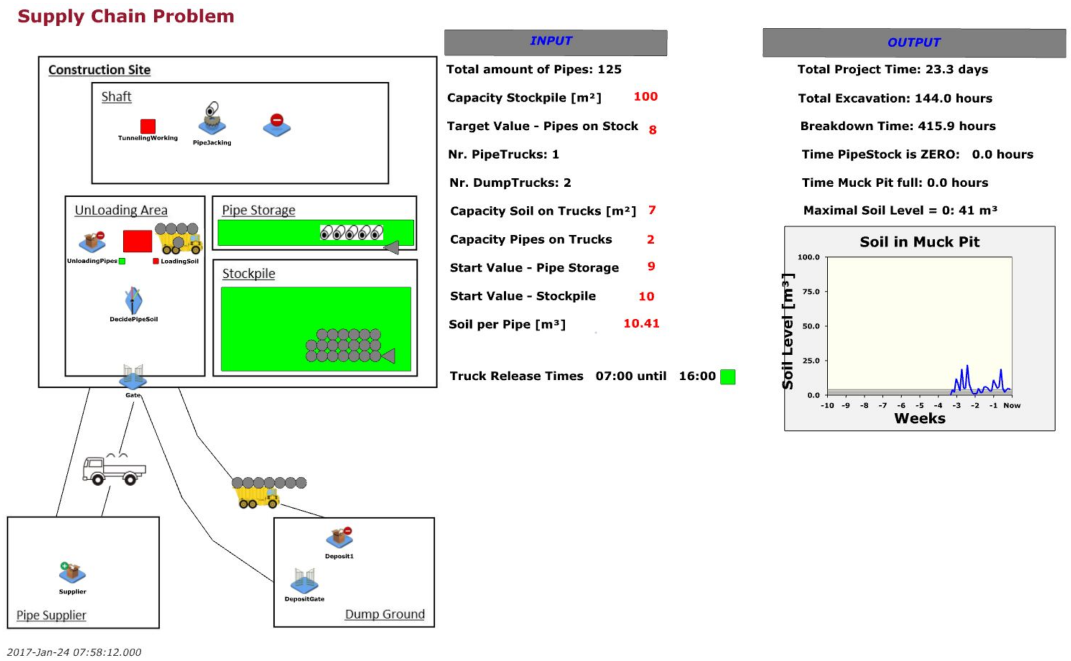


Figure 3.13: Front-end of the final computer simulation during a simulation run

3.3.1 Scenarios

One of the tasks was to define and simulate different possible order policies for the pipe and dump trucks, in order to see which one delivered the best results. The project team decided to integrate three possible scenarios:

- 5pm order policy: The number of trips, needed for the next day, are defined at 5pm depending on the soil and the pipe level.
- 7am order policy: The number of trips, needed for the same day, are defined at 7am depending on the soil and the pipe level.
- Continuous order policy: a similar approach to JIT. A trip of a truck is ordered every time, one truck load of soil (7 m^2) is produced or one load of pipes (2 pipes) is consumed.

5pm order policy Every day at 5pm the decision is made how many truck trips are needed for the next day. The calculation for ordered truck trips (t_p and t_s) is done as followed:

$$t_p = ceil(\frac{s_{pT} - s_p}{c_p}) \tag{3.1}$$

$$t_s = floor(\frac{s_s}{c_s}) \tag{3.2}$$

with

$t_{p1} = integer$, pipe truck trips ordered for day 1
$t_{s2} = integer$, dump truck trips ordered for day 1
$c_p = 2$, capacity of pipe truck
$c_s = 7$, capacity of dump truck $[m^2]$
$s_{pT} = 8$, target value for pipe storage
$s_p = \{0, 1, ..., s_{pMax}\}$, current pipe level
$s_s = \{0, 1, ..., s_{sMax}\}$, current soil level
$s_{pMax} = 10$, capacity of the pipe storage
$s_{sMax} = 100$, capacity of the muck pit (soil) $[m^2]$

The advantage of this policy is that the hired truck company and the truck driver are able to schedule their trips in advance. Complications caused by short-term orders can be avoided with that policy. On the other hand, it can lead to a higher soil level because the overnight production is excluded in this calculation.

The question arose, if a prediction of overnight progress should have been included into this policy, however in the real project shifts with zero soil production or pipe consumption were not uncommon. A predictive factor in the calculation would have led to cases where too many trips would have been ordered. This in turn would have led to additional expenses. Thus, the idea was dismissed quite quickly.

7am order policy This policy uses the same methodology as used in the 5pm policy. The decision as to how many trips are needed for the same day (day 0) is made at 7am.

The advantage of this policy is that the entire over-night production and consumption is considered in the model. On the other hand the truck drivers, whether external or internal, have to be very flexible. Indeed, each day the supply chain concerned parties have to be prepared for a utilization of between zero and a hundred percent.

Continuous order policy The third policy is a rather different approach compared to the prior two policies. There are neither fixed points in time for ordering trips, nor a fixed amount of trips included in this policy. A dump truck is ordered every time a truck load of soil is produced and parallel to that a pipe truck is ordered as soon as a truck load of pipes is consumed.

This policy results in the greatest uncertainty for the trucks. In the morning, a certain number of trips can be expected due to over-night production. Additionally to that, an undefined number of trips will be ordered during the day.

Besides the comparison of order policies, it was in the interest of the construction site how many trucks they would need for the project in order to avoid delays of the pipe jacking process. Therefore varying numbers of available truck combinations were simulated. For each scenario 20 simulation runs were performed, which the following chapter describes.

3.3.2 Results and Interpretations

This section presents the results and evaluations of the study. Each scenario was run 20 times in order to get reasonable results.

Around the construction site space is limited; the waiting area only provides space for two trucks. A third truck in the queue would have to look for a parking area near the construction site or would have to park at the roadside. Thus, cases with queue length of three or more must be avoided. Closer inspection of the construction site showed that the number of dump trucks for each day varies between zero and four. During the entire project, only one pipe truck was used to deliver pipes. However, in the simulation a second available pipe truck was added in order to see the impact of varying truck configurations to the system.

For this purpose all possible configuration of available trucks, with $x_d = \{1, 2, 3, 4\}$ dump trucks and $x_p = \{1, 2\}$ pipe trucks, were simulated. Figure 3.14 shows the average total waiting time for trucks in queue position higher than two.

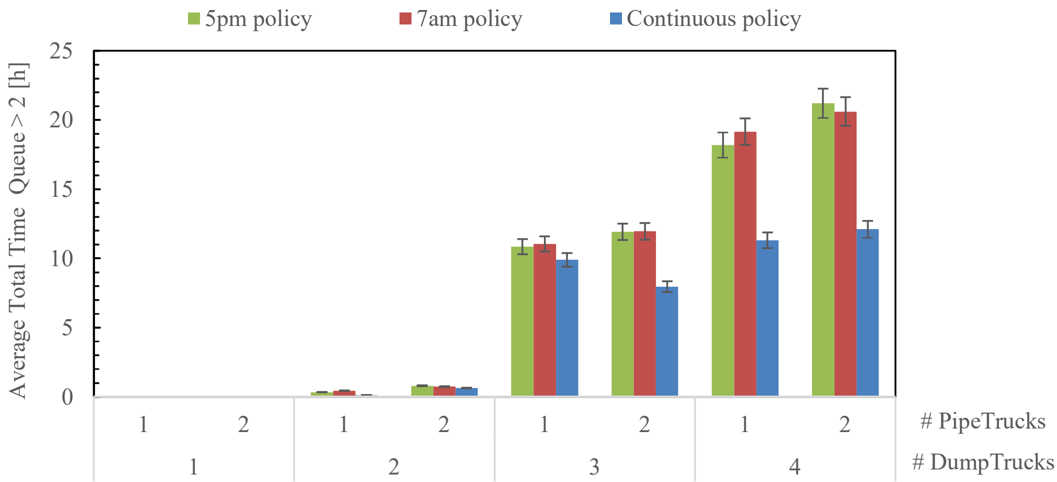


Figure 3.14: Average total time queue length is > 2

The results show that there are no waiting times with only one dump truck. Even in the worst case, when two pipe trucks and one dump truck arrive at the same time at the construction site, the queue length would never get longer than two. A second dump truck would cause up to one hour of average waiting time, depending on the chosen order policy. With a total project time of around 60 days that would still be acceptable.

There is also restricted space within the construction site. Only one truck is able to get loaded or unloaded at a time; arriving trucks have to wait in the waiting area in the meantime. This time is lost and causes delays to the supply chain. The question is, which pipe and dump truck configuration causes the least total waiting time. Figure 3.15 shows the average total waiting time for trucks.

The results for this scenario shows pretty similar results to figure 3.14. As one might expect, the average total waiting time for trucks is lowest with a low number of total trucks. However, further simulation analyses have to be made in order to show whether one dump truck is able to remove soil from the construction site fast enough without causing any delay to the pipe jacking process.

In both figures another conclusion can be drawn. The continuous order policy displays significantly better results in all cases. This can be explained by a more homogeneous

Figure 3.15: Average total waiting time for trucks at the construction site

distribution of truck arrivals over the day. With 7am and 5pm policy a bottleneck situation may occur especially in the morning which causes a longer than average waiting time and inconvenient queueing situations.

The objective of the next scenario was to show the behaviour of the soil level of the muck pit with varying dump trucks and different order policies. The pipe jacking process needs to be stopped, if the capacity of the muck pit is reached. Thus, the focus of this simulation scenario is on cases where the soil level reaches a maximum ($100m^2$). Figure 3.16 displays the maximum soil level of the entire project runtime for each scenario.

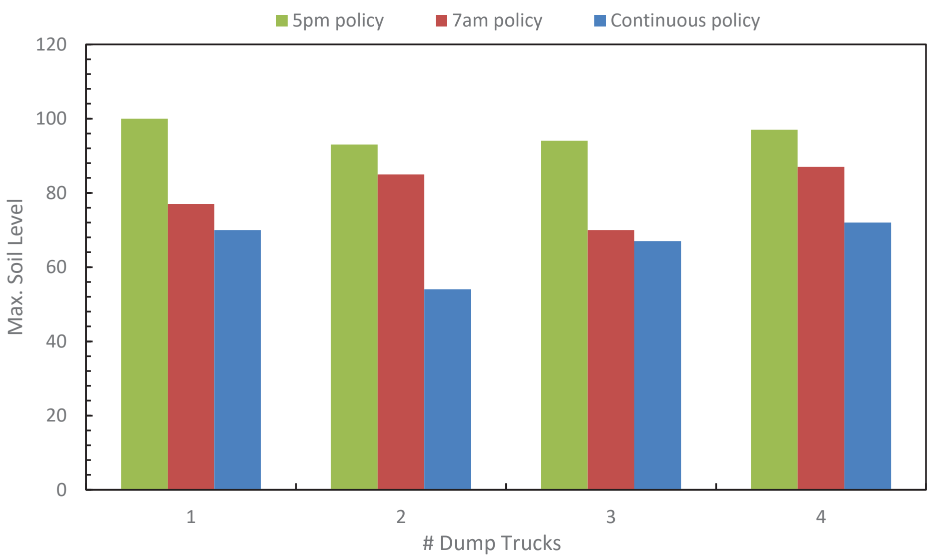


Figure 3.16: Maximum soil levels of the muck pit with varying number of dump trucks

With the results of this investigation some insights of the muck pit's behaviour could be gained. The 5pm order policy displays the highest maximum soil level among all scenarios. The maximum capacity of the muck pit is reached only once with dump truck $x_1 = 1$. This case caused a total delay of the pipe jacking of 5.20 hours. With two or more available trucks, the soil level did not reach the muck pit capacity in any case. The second insight was that the 7am order policy delivered better results than ordering the previous day. The lowest maximum soil level could be reached with the continuous order policy. However, only looking at the maximum values does not lead to meaningful interpretations of the behaviour of the muck pit. Another interpretation of the data had to be found in order to prove if the variation of single values (figure 3.16) is just random or due to other reasons.

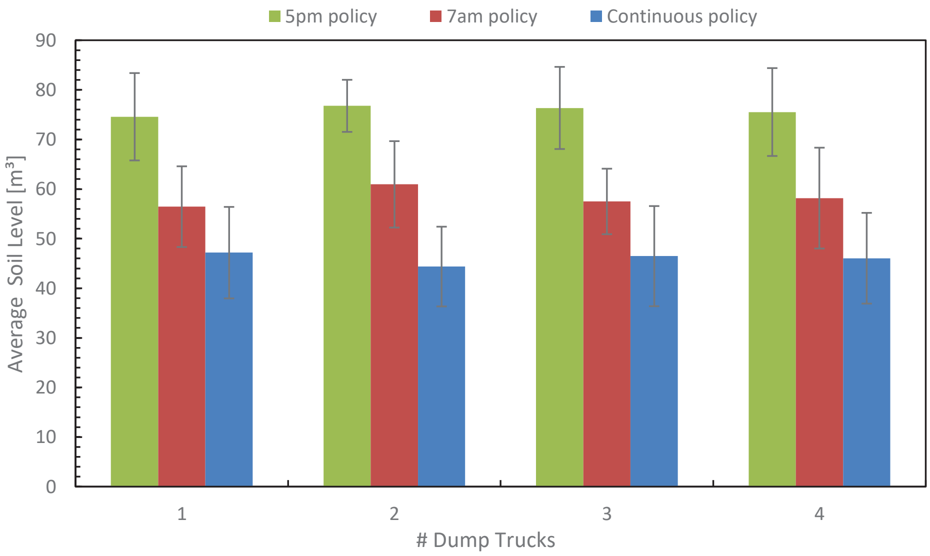


Figure 3.17: Average maximum soil levels of the muck pit with varying number of dump trucks

Figure 3.17 displays the average maximum soil level of each scenario. The graph shows similar results concerning the different order policies. A continuous order policy delivers the best results among all scenarios. However, it shows that the maximum pipe level does not change significantly with varying number of available dump trucks. In other words, an increasing the number of available dump trucks does not improve the situation at the muck pit.

The next area to look at was the pipe delivery. The key factor for assessing the pipe supply chain was the total pipe jacking delay time caused by low pipe storage level. The analysis of the delay times is displayed in figure 3.18.

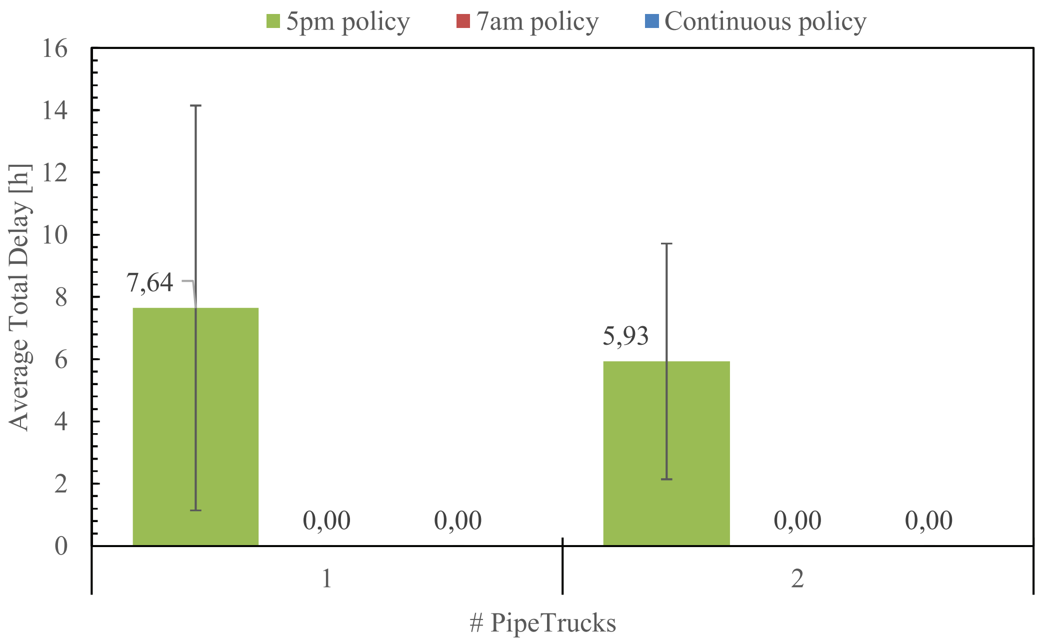


Figure 3.18: Delays of the pipe jacking process, caused by low pipe level

Obviously, there are no problems caused by the pipe supply chain in the continuous and 7am order policy. Only the policy with a truck order at 5pm causes delays to the system. This state can be traced back to the fact that with the 5pm order policies of an over-night production are not taken into consideration. More pipe trucks only marginally improve the situation.

Chapter 4

Conclusion and Outlook

This thesis deals with the development of an entire simulation study with the use of the HCCM framework and the simulation software JaamSim. In the course of a case study, logistic processes of a construction site had been analysed and simulated. The objective of the entire study was to show the usability of the HCCM framework and to point out possible improvements of the real system. This chapter concludes the findings of the case study and points out future opportunities.

Building the conceptual model was pretty straightforward. The HCCM framework delivers an excellent step-by-step instruction. The entire model could be developed well with the suggested tools and representation methods. Although the framework was introduced for more complex systems in the first place, it could be shown that it is also applicable for systems with less and rather simple control polices. Furthermore the representation of a system without queues correlates with "lean thinking" where queues represent waste and have to be avoided. An extensive part of the simulation study was the data acquisition. General information about the logistics and related processes could be gathered very well by interviewing the construction site's responsible persons. A bigger challenge was the acquisition of detailed information about the processes. Although at the construction site data of many processes were recorded in advance, the accuracy and level of detail was insufficient for the simulation study. Assumptions and simplifications had to be made to the model due to the lack of data. In the end, a conceptual model could be developed which represented the logistics situation very well.

More accurate data would lead to a more meaningful simulation study, thus, the data recording needs to be planned and coordinated for the intended purpose in the first place. In this case, one suggestion to improve this issue would be to integrate a digital check-in system for trucks at the construction site. Breakdown times could be recorded with time stamps and activity durations with vibration sensors. GPS tracker, installed in the trucks, could record travel times and potential improvements.

The HCCM framework provided a good instruction for the development of the conceptual model. The structure and all activities of the system could have been described well and the control description could emphasise the advantages of a no-queue based representation. This case study, however, could not show all major benefits of the HCCM framework. The core element of this framework is the control view of a problem. In this study, the control units were kept very simple. The greater advantages of the HCCM framework, however, can be seen especially with more complex systems with sets of decision policies or where entities change their roles. This can be examined with further simulation studies in this area.

The actual computer simulation was built based on the conceptual model. Three different order policies were simulated and compared. It shows a clear picture of the situation and the logistics' behaviour at the construction site. The best results among the scenarios were archived with the continuous order policy. It delivered the lowest average soil level of the muck skip and lowest average waiting time for trucks. With this insight, observations of the muck pit size can be made. Furthermore, with this policy it would be possible to reduce the maximal soil capacity and as a result improve the space conditions at the construction site.

But this conclusion does not compulsorily lead to a discardment of the 7am and 5pm order policy. The continuous order policy displays a very irregular order interval. This requires a very flexible truck driver (internal) or transport company (external). The next stage would be to coordinate the possible order policies with the transport companies. It needs to be evaluated whether irregular orders over one day (continuous order policy) are more favourable compared to fixed defined trips which may cause a higher truck waiting time especially in the morning (7am and 5pm order policy). What would be the best and most convenient solution for this situation? Integration of truck rosters (morning and evening shifts) in the simulation could the first step in order to answer this question.

The computer simulation can be used as a tool for planning future projects. Varying soil production and pipe consumption, as well as the tunnel length can be adopted. The outcome of the new simulation is the number of necessary available trucks. With the number of trucks it is possible to plan the size of the loading and waiting area.

In general, the construction sector is faced with many challenges in the future. Due to globalization, competition is increasing strongly. Thus, companies are faced with a high volatile environment. In order to stay competitive, subjects such as agility and flexibility have to be addressed.

Fast changing circumstances are in the nature of construction projects. Whereas in production industry many activities are processed repeatedly and in many cases automatically, in the field of construction management characteristics of processes may vary strongly among a project.

Manufacturing industry provides a set of ideas and concepts for improving the agility and performance of a company. The concept of lean management has already been discussed broadly for the field of construction. Concepts such as Just in Time (JIT), bottleneck analyses or Total Productive Maintenance (TPM) provide approaches for improving the entire efficiency. In the author's opinion, the implementation of those ideas still lags behind the state of research, however. Developing simulation studies can also help getting a better understanding of the activities and behaviour of the system. For that reason alone, companies have to work on this issue with greater efforts. Whereas simulation studies are very successful in research, they did not find much acceptance in construction management, yet. The biggest challenge will be how to deal and implement these mentioned ideas into the environment of construction management. Researchers have to focus in particular on tools and approaches that are feasible in practice as well.

On the other hand, the production industry is faced with an increasing degree of volatility as well. A different approach for finding new ideas in the manufacturing sector would be to look into the field of construction. Since, the construction branch has a long history of uncertainty in projects, there might be ideas, which could be transferred to production. In the opinion of the author, it should be questioned, whether having a detailed plan for every scenario leads to the best solution. Over-planning very often results in rigid structures with less possibilities for influences. One idea could be to integrate the "human factor" more strongly within the production. Exceptional situations could be handled individually by humans, based on their expert knowledge and less by prescribed actions, like it is the case in construction sector.

Appendix

Conceptual Model

Problem Description

The New Zealand organization Auckland Transport is improving the public transport situation by building an underground rail line within the central business district of Auckland. As part of this project, storm water pipes have to be replaced. Pipe-jacking, a trenchless method for installation of underground pipes, is the chosen method for this operation. This model represents the current supply chain problem of one specific construction site. The problem consists two main activities, pipe supply and soil removal.

Pipe Supply One pipe truck (capacity 2 pipes) delivers pipes from the supplier to the construction site. The quantity of delivered pipes depends on the advancement of the pipe-jacking process and varies between zero and six pipes, respectively zero and three trips per day. The main sequence of the pipe supply can be described as followed. If pipes are requested, the truck is gets loaded at the pipe supplier. After the loading process the truck goes to the construction site. This travel duration varies depending on the traffic situation. After arriving at the construction site, the pipes are unloaded by the gantry crane. The crane is not available at all times, thus the truck has to wait until the gantry crane is available. After unloading the pipes, the truck goes back to the supplier. Ideally, this procedure is repeated up to three times a day, at 10:00 am, 1:00 pm and at 16:00 pm. However, deliveries can be cancelled due to changing demand caused by delays in the pipe-jacking process.

Soil Removal Depending on the advancement of the pipe-jacking process, different amounts of soil are produced and must be stored in the muck pit. When emptying of the muck pit is necessary, one or more dump trucks are ordered. After arriving at the construction site an excavator loads the truck. The fully loaded truck transports the soil to the dumping ground. This travel duration varies depending on the traffic situation.

This procedure is repeated during the operating hours for each truck until the muck pit is empty. Further constraints define the loading and unloading process at the construction site. Only one truck can be served (loading soil or unloading pipes) at a time. There is room for a maximum one additional truck, waiting to be served. The process of unloading pipes has the highest priority, however, the loading process of a dump truck cannot be interrupted.

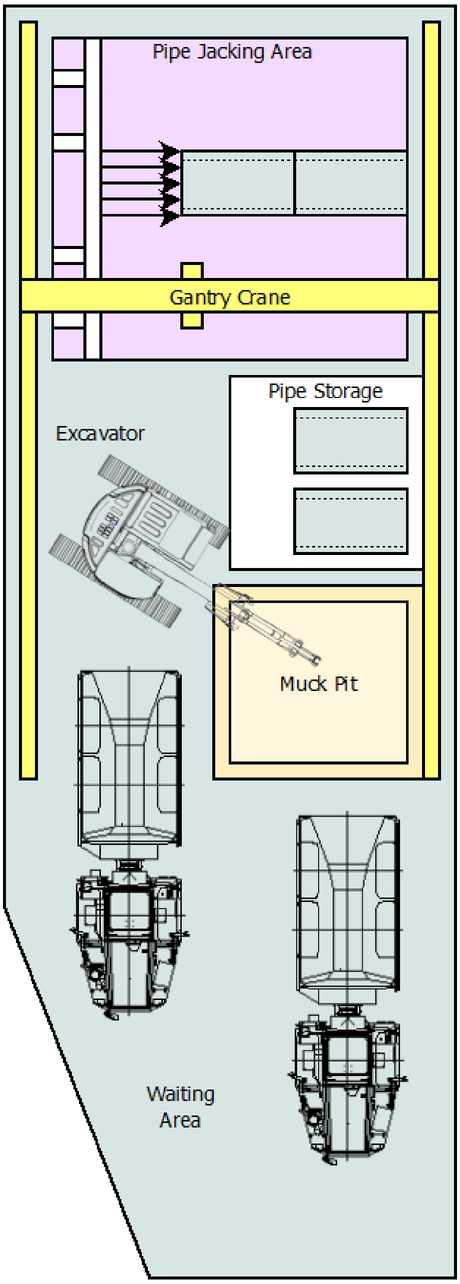


Figure A.1: Layout of the Construction Site

Table A.1: Simulation Study Objectives

Organizational Aim	
No interruption of the pipe-jacking process due to problems on the surface (transport or storage)	
General Objectives	
Project duration	4 months
Workload	1 Developer
Visualization requirements	Visualization of the computer model with JaamSim
Reusability requirements	The model should be designed with respect to changing conditions for future transport problems.
Documentation and usability	The simulation should be easy to handle and easy to understand, even for untrained personnel. A detailed documentation is required.
Modelling Objectives	
Policy improvements	Evaluating different policies for truck utilisation including ordering.

Table A.2: Simulation Outputs

Aggregated Values	
Mean waiting time - trucks	Average time pipe or dump trucks have to wait to enter the construction site
Total time of delay	Delays of the pipe-jacking process caused by surface problems
Total simulation time	The total time of the simulation
Max. soil level	The maximum level of soil in muck pit

Table A.3: Experimental Factors

Experimental Factors	
Capacity muck pile	Capacity of the muck pile at the construction site
Target value pipes	Pipe storage target value at the construction site
# Pipe trucks	Number of trucks available for pipe transport
# Dump trucks	Number of trucks available for transport soil to dumping ground
Travel time - supplier	Travel time distribution from and to supplier - Google Directions data
Travel time - dumping ground	Travel time distribution from and to dumping ground - Google Directions Data

Table A.4: Simulation Inputs

Input Factors	
Total # pipes	Total number of pipes
Soil per pipe	Average soil issued per pipe
Breakdown times	Breakdown duration and interval of the pipe jacking process
Construction site operating hours	Net time on which the construction site is working
Operating hours	Starting and end daytime of truck release
Duration for loading/unloading soil	Durations for un/loading soil at the construction site and dump ground
Duration for loading/unloading pipes	Durations for un/loading soil at the construction site and supplier

Structural View of the Problem

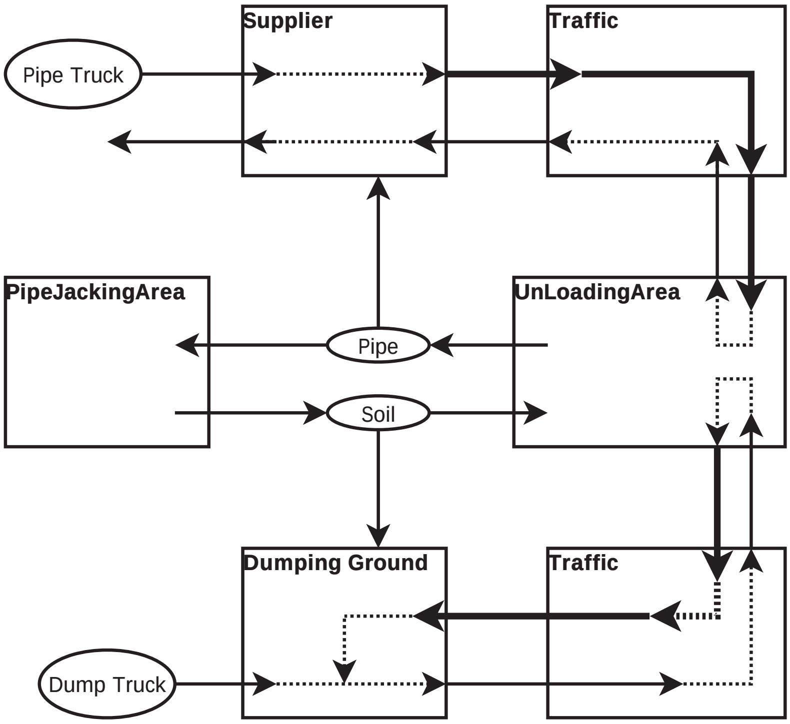


Figure A.2: Structure of the supply chain (thick line represents loaded conveyance)

Model Scope

Table A.5: Entities within the Simulation Model

Entities		
Component	**In/Exclude**	**Justification**
Truck	Include	Entity container - required flow entity
- DumpTruck	Include	Entity container for Soil - required flow sub-entity
- PipeTruck	Include	Entity container for Pipes - required flow sub-entity
UnLoadingArea	Include	Contains all loading and unloading activities as well as access polices
PipeJackingArea	Include	Tunnelling operation, consumes Pipes and produces Soil
Traffic	Include	Contains traffic data - modelled as delay for Trucks
- Traffic Supplier	Exclude	Traffic from and to Supplier of pipes
- Traffic Dumping	Exclude	Traffic from and to Dumping Ground
Dumping Ground	Exclude	Is represented by the Soil Unloading activity
Supplier	Exclude	Is represented by the Pipe Loading activity
Gantry Crane	Exclude	UnLoading Area represents behaviour of Gantry Crane
Excavator	Exclude	UnLoading Area represents behaviour of Excavator
Pipe	Exclude	Handled as attribute of other entities
Soil	Exclude	Handled as attribute of other entities
Construction Site	Exclude	Represented by UnLoadingArea and PipeJackingArea
Waiting Area	Exclude	Represented by UnLoadingArea

Table A.6: Activities within the Simulation Model

Activities		
Component	**In/Exclude**	**Description**
Waiting at Construction Site	Include	Activity for PipeTruck and DumpTruck
Soil Loading (UA)	Include	Loading soil from muck pit onto DumpTruck at the UnLoadingArea
Soil Loading (Truck)	Include	Activity of DumpTruck during Soil Loading (UA)
Pipe Unloading (UA)	Include	Unloading PipeTrucks at UnLoadingArea
Pipe Unloading (Truck)	Include	Activity of the PipeTruck during Pipe Loading (UA)
Driving	Include	The activity of Trucks being in Traffic
Pipe Loading	Include	Loading pipes onto Pipe Truck
Soil Unloading	Include	Unloading of DumpTruck at the deposit ground
Waiting for Pipe Loading	Include	PipeTruck waiting for pipes getting loaded
Pipe Jacking	Include	Includes all activities from pipe consumption (input) to soil production (output)
Gantry Crane	Exclude	Unloads PipeTruck and transports Soil from pipe jacking area to muck pit (out of scope)

Table A.7: Control Units within the Simulation Model

Control Units		
Component	**In/Exclude**	**Justification**
Pipe Jacking	Include	Controls the Pipe Jacking as interaction between UnLoading Area and Pipe Jacking Area
Truck Access	Include	Controls the access to the UnLoading Area
Truck Order	Include	Manages order for needed pipe deliveries and soil dumping

Table A.8: Entity Structure and Attributes

Entity Attributes				
Component	**Act./Pas.**	**Attributes**	**Unit**	**Constant Value**
Truck	Active	Truck.Type Truck.Load Truck.WaitingTime Truck.AggregatedWaitingTime	Pipe OR Dump varies (m^2 or #) min min	
DumpTruck	Active	DumpTruck.Capacity DumpTruck.RouteToDumpground	m^3 boolean	
DumpTruck	Active	PipeTruck.Capacity PipeTruck.RouteToSupplier	# boolean	
UnLoadingArea(UA)	Active	UA.AggregatedWaitingTime UA.Trucks UA.QueueLength UA.TimeUnloadingPipes UA.TimeLoadingSoil UA.PipeLevel UA.SoilLevel UA.SoilCapacity UA.PipeCapacity	min # # min min # m^2 m^2 #	
PipeJackingArea	Active	PipeJackingArea.Operating PipeJackingArea.TotalPipes PipeJackingArea.SoilPerPipe	boolean # #	
Traffic	Passive	Traffic.DurationToSupplier Traffic.DurationFromSupplier Traffic.DurationToDumpGround Traffic.DurationFromDumpground	min min min min	

Activities

Table A.9: "Waiting at Construction Site" Activity

Waiting at Construction Site	
Participating Entities	DumpTruck, PipeTruck UnLoadingArea
Start Type	Sequential
End Type	System

Start State Changes	# UA.Trucks += 1
	# UA.QueueLength += 1
	StartTime = time

End State Changes	# UA.QueueLength -= 1 Truck.AggregatedWaitingTime += time – StartTime Truck. WaitingTime =time – StartTime
Attributes	**Description/Value**
StartTime	Time activity starts

Table A.10: "Soil Loading (UA)" Activity

Soil Loading (UA)	
Participating Entities	UnLoadingArea, DumpTruck
Start Type	Requested
End Type	Scheduled
Start State Changes	none

End State Changes	UA.SoilLevel -= DumpTruck.Capacity
	# UA.Trucks -= 1
	DumpTruck.RouteToDumpingGround = TRUE

Attributes	**Description/Value**
Duration	Input data [min]
DumpTruck.Capacity	Input data [m^3]
Request Attributes	**Description/Value**
TimeRequest	When was the request made from Truck Access Control Unit
Request Specification	Request filed after Soil Loading

Table A.11: "Soil Loading (Truck)" Activity

Soil Loading (Truck)	
Participating Entities	UnLoadingArea, DumpTruck
Start Type	Requested
End Type	Scheduled
Start State Changes	none
End State Changes	DumpTruck.RouteToDumpingGround = TRUE
Attributes	**Description/Value**
Duration	Duration of activity "Soil Loading (UA)"
Request Attributes	**Description/Value**
TimeRequest	When was the request made from UA
Request Specification	Request filed after Soil Loading

Table A.12: "Pipe Unloading" Activity at UnLoadingArea

Pipe Unloading(UA)	
Participating Entities	UnLoadingArea, PipeTruck
Start Type	Requested
End Type	Scheduled
Start State Changes	none
End State Changes	UA.PipeLevel -= PipeTruck.Capacity # UA.Trucks -= 1
Attributes	**Description/Value**
Duration	Input data [min]
PipeTruck.Capacity	Input data [#]
Request Attributes	**Description/Value**
TimeRequest	When was the request made from Truck Access Control Unit
Request Specification	Request filed after Pipe Unloading

Table A.13: "Pipe Unloading" Activity for Truck entity

Pipe Unloading(Truck)	
Participating Entities	UnLoadingArea, PipeTruck
Start Type	Requested
End Type	Scheduled
Start State Changes	none
End State Changes	PipeTruck.RouteToSupplier = TRUE
Attributes	**Description/Value**
Duration	Duration of activity "Pipe Unloading(UA)"
Request Attributes	**Description/Value**
TimeRequest	When was the request made from UA
Request Specification	Request filed after Pipe Unloading

Table A.14: "Driving" Activity for Truck entity

Driving	
Participating Entities	Truck, PipeTruck, DumpTruck, Traffic
Start Type	Sequential
End Type	Scheduled
Start State Changes	none
End State Changes	PipeTruck.RouteToSupplier = FALSE DumpTruck.RouteToDumpingGround = TRUE
Attributes	**Description/Value**
Duration	Duration of activity "Pipe Unloading(UA)"
Weekday	Current weekday
Time	Actual daytime [hh:mm]

Table A.15: "Waiting for Pipe Loading" Activity for Truck entity

Waiting for Pipe Loading	
Participating Entities	PipeTruck
Start Type	Sequential
End Type	Requested
Start State Changes	StartTime = time
End State Changes	Truck.AggregatedWaitingTime += time – StartTime Truck. WaitingTime =time – StartTime
Attributes	**Description/Value**
StartTime	Time activity starts
Request Attributes	**Description/Value**
TimeRequest	When was the request made by Truck Order Control Unit
Request Specification	Request filed after end of PipeJacking

Table A.16: ”Pipe Jacking” Activity for PipeJacking entity

PipeJacking	
Participating Entities	UnLoadingArea, Pipe JackingArea
Start Type	Scheduled
End Type	Requested
Start State Changes	PipeJackingArea.Operating = TRUE
End State Changes	PipeJackingArea.Operating = FALSE
Attributes	**Description/Value**
Duration	Duration defined by breakdown data
Request Attributes	**Description/Value**
TimeRequest	When was the request made by breakdown data
Request Specification	Request filed after end of PipeJacking

Table A.17: Control Unit Structure and Definition

Control Units		
Name	**Entities**	**Attributes**
PipeJacking	PipeJacking Area	PipeJackingArea.Operating
	UnLoadingArea	UA.PipeLevel
		UA.SoilLevel
Truck Access	Truck	Truck.Type
	UnLoadingArea	# UA.Trucks
Truck Order	UnLoadingArea	UA.PipeLevel
		UA.SoilLevel
		# TruckOrder.PipeTruck
		# TruckOrder.DumpTruck
	DumpTruck	DumpTruck.Capacity
	PipeTruck	PipeTruck.Capacity

Table A.18: Assumptions and Simplifications within the Model

Assumptions	**Confidence**	**Impact**
Same breakdown rates for future projects	Medium	High
Simplifications		
No severe traffic breakdowns in model considered	Low	Low
PipeJacking and UnLoading activities do not influence each other directly (exclude gantry crane)	Low	Medium
Exclude day 1 in Simulation (day 1 is represented by soil/pipe start value)	Medium	Low
All truck are from the same operator (actually 1 external and 1 internal)	Medium	Low
No "Waiting for Unloading" at DumpingGround	High	Low
No delay at Supplier caused by loading queue	Medium	Low
No fuel refill necessary	Low	Low
Pipe storage down the shaft not considered	Medium	Medium

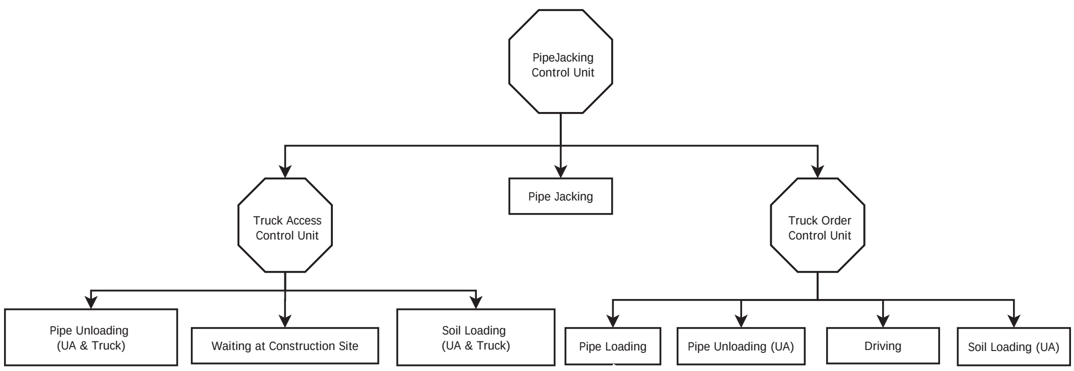


Figure A.3: Control view of the problem

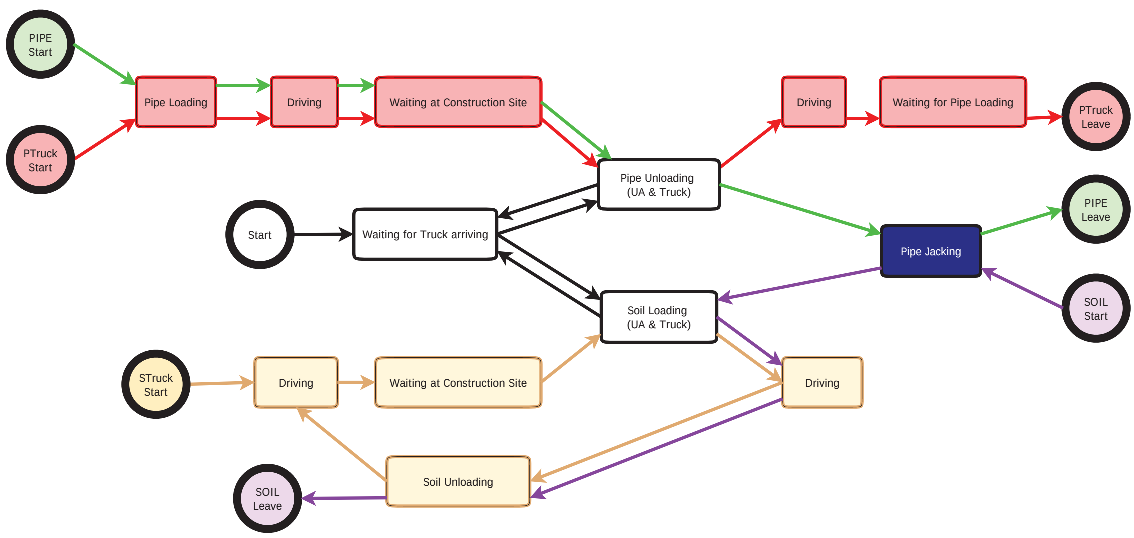
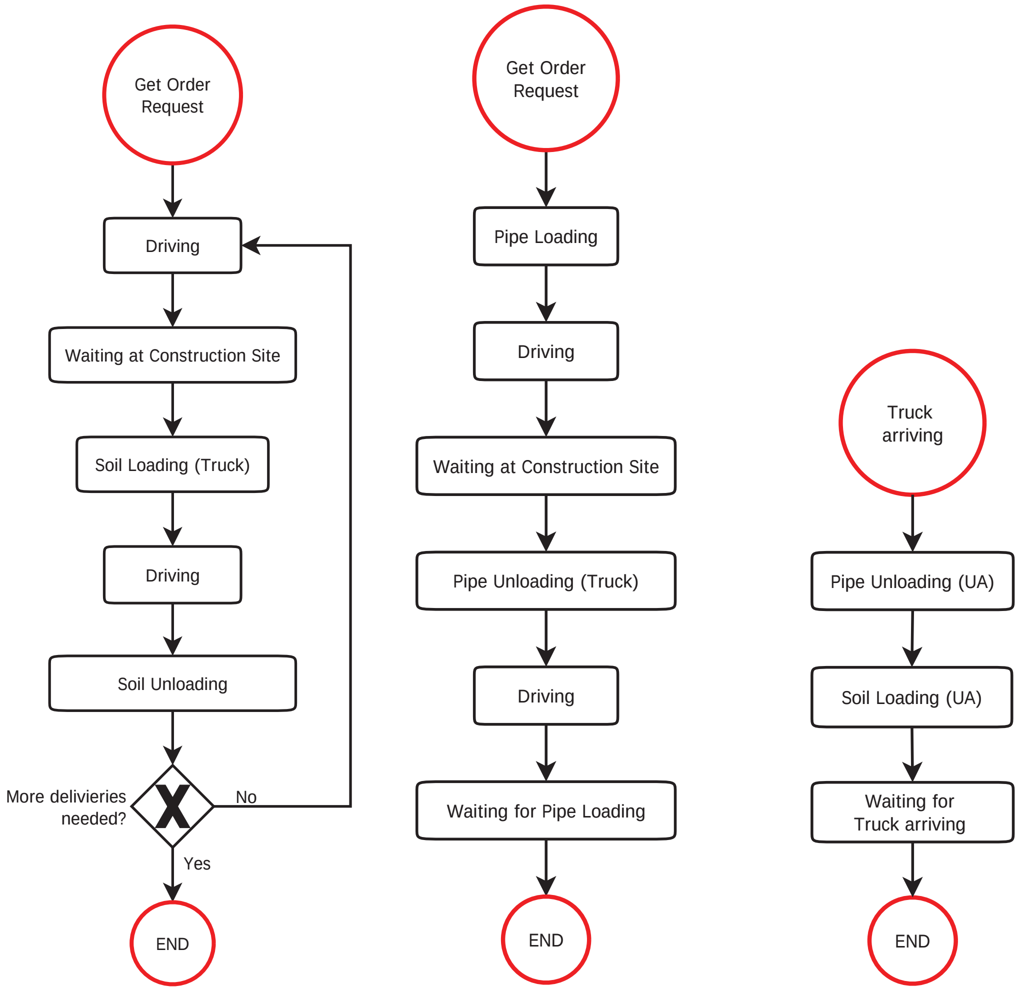


Figure A.4: Overview of entity flow in system, supplemented with pipes and soil



(a) DumpTruck (b) PipeTruck (c) UnLoadingArea

Figure A.5: Behaviour of entities

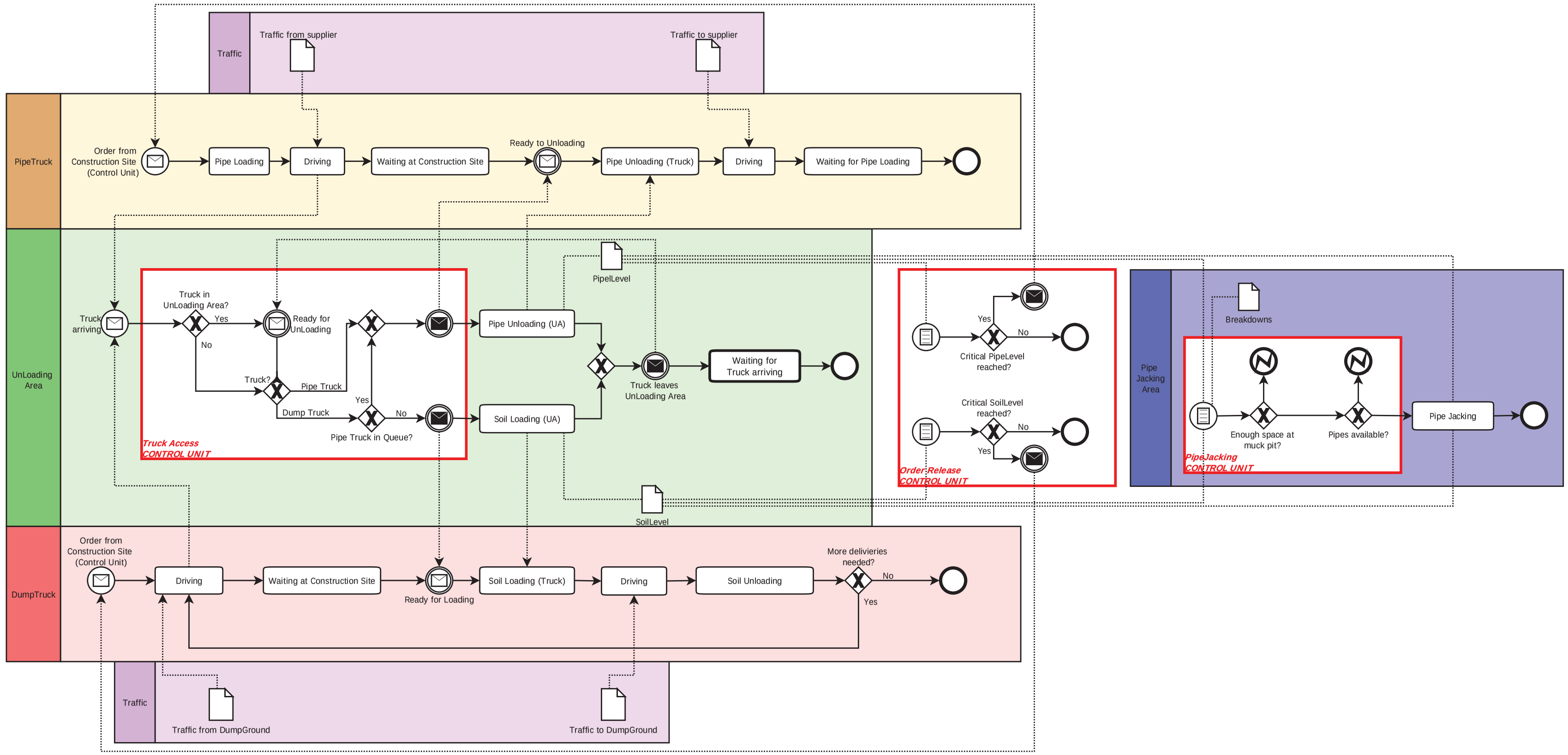


Figure A.6: Behaviour of UnLoadingArea entity

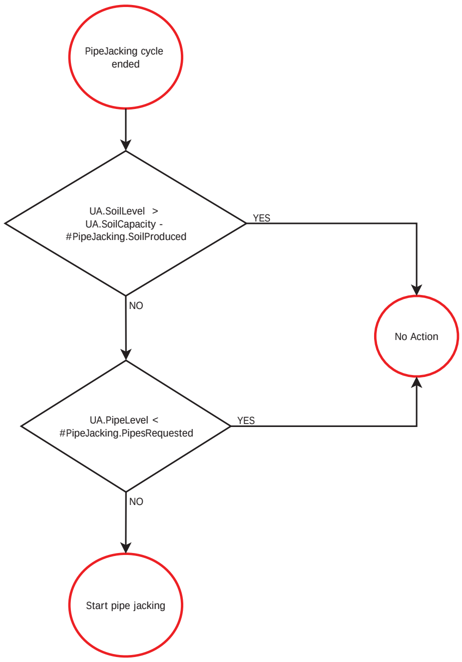


Figure A.7: PipeJacking Control Unit rules

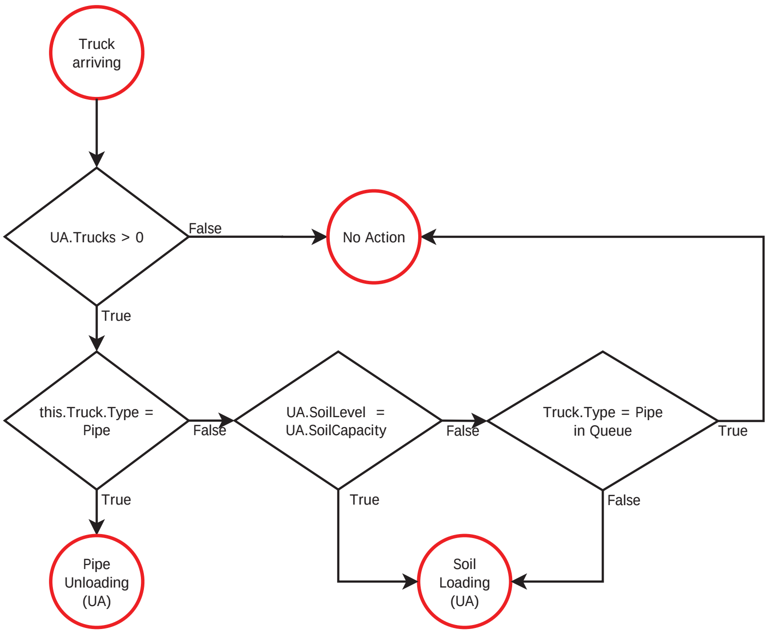


Figure A.8: TruckAccess Control Unit rules

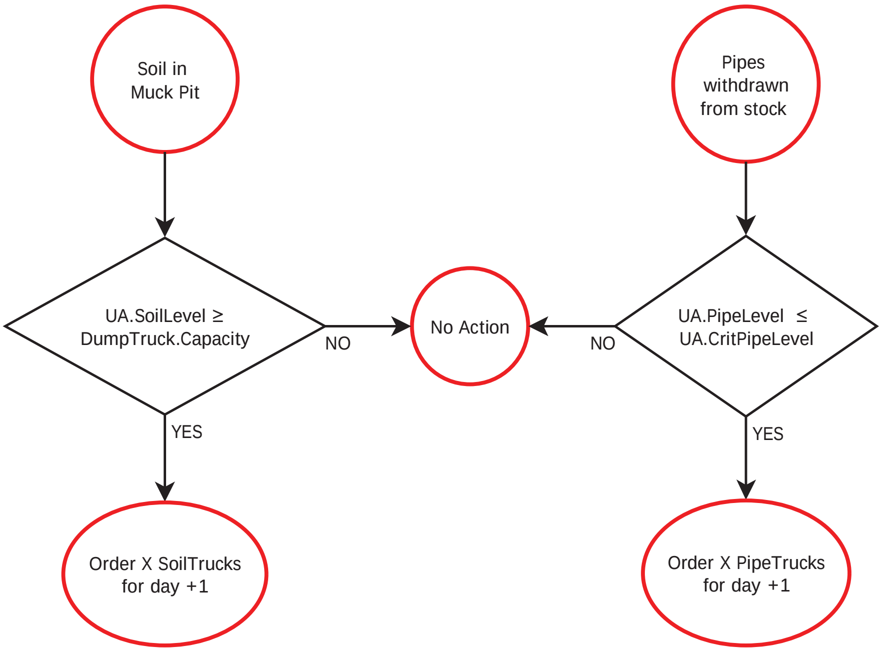


Figure A.9: TruckOrder Control Unit rules

Bibliography

AbouRizk, S. (2010a). “Role of Simulation in Construction Engineering and Management.” In: *Journal of Construction Engineering and Management-Asce* 136.10, pp. 1140–1153 (cit. on p. 36).

AbouRizk, Simaan (2010b). “Role of Simulation in Construction Engineering and Management.” In: *Journal of Construction Engineering and Management* 136.10, pp. 1140–1153 (cit. on p. 1).

AbouRizk, Simaan M. and Dany Hajjar (1998). “A framework for applying simulation in construction.” In: *Canadian Journal of Civil Engineering* 25.3, pp. 604–617 (cit. on p. 36).

Balci, O. (1994). “Validation, Verification, and Testing Techniques Throughout the Life Cycle of a Simulation Study.” In: *1994 Winter Simulation Conference Proceedings*, pp. 215–220 (cit. on p. 12).

Balci, O. and W F Ormsby (2007). “Conceptual modelling for designing large-scale simulations.” In: *Journal of Simulation* 1.3, pp. 175–186 (cit. on pp. 12–15, 20).

Banks, Jerry et al. (2004). *Discrete-event system simulation*. Prentice-Hall international series in industrial and systems engineering. Englewood Cliffs, NJ u.a.: Prentice-Hall, XIV, 514 S. (Cit. on pp. 6, 10, 27, 28, 33).

Birta, Louis G. and Gilbert Arbez (2007a). *ABCmod: a conceptual modelling framework for discrete event dynamic systems*. Conference Paper. San Diego, California (cit. on p. 19).

Birta, Louis G. and Gilbert Arbez (2007b). *Modelling and simulation : exploring dynamic system behaviour*. London: Springer, xvii, 454 p. (Cit. on pp. 13, 16, 19, 21).

Borshchev, A. and A. Filippov (2004). “From System Dynamics and Discrete Event to Practial Agent Based Modeling: Reasons, Techniques, Tools.” In: *The 22nd International Conference of the System Dynamics Society* (cit. on pp. 7, 8).

Box, George EP (1979). “Robustness in the strategy of scientific model building.” In: *Robustness in statistics* 1, pp. 201–236 (cit. on p. 16).

Brooks, R. J. and S. Robinson (2000). "Simulation Studies: Key Stages and Processes." In: *Palgrave Macmillan* (cit. on pp. 5, 31).

Cassandras, C. G. and S. Lafortune (1999). *Introduction to discrete event systems.* Kluwer international series on discrete event dynamic systems. Boston: Kluwer Academic, p. 822 (cit. on p. 11).

Cox, A., P. Ireland, and M. Townsend (2006). *Managing in Construction Supply Chains and Markets.* Report. Thomas Telford (cit. on p. 38).

Deng, J. L. (1982). "Control-Problems of Grey Systems." In: *Systems and Control Letters* 1.5, pp. 288–294 (cit. on p. 30).

Furian, N. et al. (2015). "A conceptual modeling framework for discrete event simulation using hierarchical control structures." In: *Simul Model Pract Theory* 56, pp. 82–96 (cit. on pp. 2, 13, 16, 19–23, 40).

Gass, S. I. (1984). "Documenting a Computer-Based Model." In: *Interfaces* 14.3, pp. 84–93 (cit. on p. 32).

Halpin, D. W. (1977). "Cyclone - Method for Modeling Job Site Processes." In: *Journal of the Construction Division-Asce* 103.3, pp. 489–499 (cit. on p. 36).

Hedtstück, U. (2013). *Simulation diskreter Prozesse.* Berlin Heidelberg: Springer Vieweg, pp. XIII, 235 (cit. on p. 9).

JaamSim, Development Team (2016). *JaamSim: Discrete-Event Simulation Software.* Manual (cit. on p. 34).

Karagöz, N Alpay and Onur Demirörs (2011). "Conceptual modeling notations and techniques." In: *Conceptual modeling for discrete-event simulation*, pp. 179–209 (cit. on pp. 13, 16, 17).

Kleijen, J. P. C. (1999). *Validation of models: statistical techniques and data availability.* Conference Paper (cit. on pp. 26–28).

Koskela, L. (1992). *Application of the New Production Philosophy to Construction.* Report (cit. on p. 36).

Koskela, L. (1993). "Lean Production in Construction." In: (cit. on p. 36).

Kotiadis, K. and S. Robinson (2008). "Conceptual Modelling: Knowledge Acquisition and Model Abstraction." In: *2008 Winter Simulation Conference, Vols 1-5*, pp. 951–958 (cit. on pp. 13, 14, 16).

Lacy, L.W., S. Youngblodd, and R. Might (2001). "Developing a Consensus Perspective on Conceptual Models for Simulation Systems." In: *Proceedings of Spring Simulation Interoperability Workshop* (cit. on p. 12).

Landry, M., J. L. Malouin, and M. Oral (1983). "Model Validation in Operations-Research." In: *European Journal of Operational Research* 14.3, pp. 207–220 (cit. on p. 5).

Law, A. M. (2005). "How to build valid and credible simulation models." In: *Proceedings of the 2005 Winter Simulation Conference, Vols 1-4*, pp. 24–32 (cit. on p. 24).

Law, Averill M. and W. David Kelton (1991). *Simulation modeling and analysis*. 2. McGraw-Hill series in industrial engineering und management science. New York u.a.: McGraw-Hill, XXII, 759 S. (Cit. on pp. 6, 7, 10).

Liu, S. and Y. Lin (2010). *Grey Systems, Theory and Applications*. Berlin Heidelberg: Springer-Verlag (cit. on p. 30).

Martinez, J. C. and P. G. Ioannou (1994). *General purpose simulation with Stroboscope*. Conference Paper (cit. on p. 36).

Nance, Richard E. et al. (1994). *The conical methodology and the evolution of simulation model development*. Technical report. Blacksburg, Va.: Systems Research Center, Dept. of Computer Science, Virginia Polytechnic Institute, and State University, 50 p. (Cit. on p. 12).

Onggo, B S S and J. Hill (2014). "Data identification and data collection methods in simulation: a case study at ORH Ltd." In: *Journal of Simulation* 8.3, pp. 195–205 (cit. on pp. 24, 25, 27).

Oscarsson, J. and M. U. Moris (2002). "Documentation of discrete, event simulation models for manufacturing system life cycle simulation." In: *Proceedings of the 2002 Winter Simulation Conference, Vols 1 and 2*, pp. 1073–1078 (cit. on p. 32).

Pace, D. K. (2000). "Ideas about simulation conceptual model development." In: *Johns Hopkins Apl Technical Digest* 21.3, pp. 327–336 (cit. on p. 12).

Page, B. (1991). *Diskrete Simulation. Eine Einftührung mit Modula-2*. Springer-Verlag Berlin Heidelberg (cit. on p. 31).

Pidd, M. (2003). "Tools for Thinking—Modelling in Management Science." In: *Journal of the Operational Research Society* 48.11, pp. 1150–1150 (cit. on pp. 25, 26).

Pipe-Jacking-Association, International (2017). *An introduction to pipe jacking and microtunnelling*. Report (cit. on p. 39).

Rabelo, L. et al. (2005). "Enterprise simulation: A hybrid system approach." In: *International Journal of Computer Integrated Manufacturing* 18.6, pp. 498–508 (cit. on pp. 7, 8).

Robinson, S. (1994). "Simulation Projects - Building the Right Conceptual-Model." In: *Industrial Engineering* 26.9, pp. 34–36 (cit. on p. 15).

Robinson, S. (2008a). "Conceptual modelling for simulation Part I: definition and requirements." In: *Journal of the Operational Research Society* 59.3, pp. 278–290 (cit. on pp. 12, 13, 19–21).

Robinson, S. (2008b). "Conceptual modelling for simulation Part II: a framework for conceptual modelling." In: *Journal of the Operational Research Society* 59.3, pp. 291–304 (cit. on pp. 16–18, 20).

Robinson, Stewart (2004). *Simulation the practice of model development and use.* Chichester: Wiley, XIX, 316 S. (Cit. on pp. 6, 15, 24–26, 28, 29, 32–34).

Schary, B.P. and T. Skjott-Larsen (2001). "Managing the Global Supply Chain." In: *Copen Hagen Business School Press* (cit. on p. 36).

Shewchuk, J. P. and T. C. Chang (1991). *An approach to object-oriented discrete-event simulation of manufacturing systems.* Conference Paper (cit. on p. 36).

Sudnikovich, W. P. et al. (2004). "Extensible battle management language as a transformation enabler." In: *Simulation-Transactions of the Society for Modeling and Simulation International* 80.12, pp. 669–680 (cit. on p. 17).

Trybula, W. J. (1995). "Improving the modeling process." In: *1995 Ieee International Conference on Systems, Man and Cybernetics, Vols 1-5*, pp. 201–205 (cit. on p. 24).

Voigtmann, J.K. and H.J. Bargstädt (2010). "Simulation von Logistikstrategien im Bauwesen." In: *KIT Scientific Publishing 2010* (cit. on p. 37).

Vrijhoef, R. and L. Koskela (1999). *Roles of Supply Chain Management in Construction.* Report. University of California (cit. on p. 37).

Wainer, G. A. (2009). "Discrete-Event Modeling and Simulation: A Practitioner's Approach." In: *Discrete-Event Modeling and Simulation: A Practitioner's Approach*, pp. 1–494 (cit. on pp. 5, 8, 9).

Zeigler, B. P. (1987). "Hierarchical, Modular Discrete-Event Modeling in an Object-Oriented Environment." In: *Simulation* 49.5, pp. 219–230 (cit. on p. 12).