CHALMERS
UNIVERSITY OF TECHNOLOGY

Optimisation in Discrete-event simulation models

Identifying potential use-cases and impacts of using optimisation in Discrete-event simulation

Master's thesis in Production Engineering

ERIK FORNE ADMARKER
HERMAN VÄSTSÄTER

INDUSTRIAL AND MATERIALS SCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Optimisation in Discrete-Event simulation models

Identifying potential use-cases and impacts of using optimisation in Discrete-event simulation

Erik Forne Admarker
Herman Västsäter

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Industrial and Materials Science
Chalmers University of Technology
Gothenburg, Sweden 2024

Optimisation in Discrete-Event simulation models
Identifying potential use-cases and impacts of using optimisation in Discrete-event simulation
Erik Forne Admarker
Herman Västsäter

Supervisor: Niklas Palm, AFRY
Supervisor: Arpita Chari, Department of Industrial and Materials Science
Examiner: Anders Skoogh, Department of Industrial and Materials Science

Master's Thesis 2024
Department of Industrial and Materials Science
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Optimisation in Discrete-Event simulation models
Identifying potential use-cases and impacts of using optimisation in Discrete-event simulation
Erik Forne Admarker, Herman Västsäter
Department of Industrial and Materials Science
Chalmers University of Technology

Abstract

Discrete-event simulation is a tool used to simulate and analyse complex real-world systems. As a system gets more complex the number of parameter variations can increase, making it difficult for the user to accurately predict an optimal solution. Optimisation methods are mathematical algorithms that are specialised in finding optimal solutions for a given problem. A combination of discrete-event simulation and optimisation methods could result in a more efficient problem-solving process which has the possibility of providing more optimal solutions. This thesis uses a case study methodology that includes a literature study and a qualitative study to assess the possibilities of using optimisation tools in a simulation setting in an organisation. Optimisation tools in discrete-event simulation software exists in several available software and all of the ones studied in this thesis utilise evolutionary algorithms for optimisation calculations. This thesis handles the importance of finding the right use-case for the implementation of simulation optimisation in a project. A decision tree was created to help users navigate questions one could ask themselves when deciding whether optimisation is suitable in their projects or not.

Keywords: discrete-event simulation, optimisation, simulation, evolutionary algorithm, simulated annealing, logistic simulation

Acknowledgements

We are deeply thankful to the Supply Chain Management section of AFRY Gothenburg for allowing us to write this thesis with the insight into the section. Special thanks to the simulation team at AFRY Gothenburg and Niklas Palm, our supervisor at AFRY, for your invaluable insights and knowledge as well as the much appreciated inputs during the entirety of this thesis.

We are also particularly grateful to all of the individuals who participated in interviews and discussions that provided us with the resources necessary to complete this thesis.

We extend our heartfelt gratitude to our respective families for their unwavering support and encouragement throughout our academic journey and the completion of this Master's thesis.

We would also like to express our sincerest appreciation and gratitude towards our supervisor at Chalmers University of Technology, Arpita Chari, and our examiner Anders Skoogh for their invaluable guidance, support, and encouragements throughout the duration of this Master's thesis.

Erik Forne Admarker, Herman Västsäter, Gothenburg, June 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AI	artificial intelligence
CMA-ES	covariance matrix adaptation evolution strategy
DES	discrete-event simulation
EA	evolutionary algorithms
GA	genetic algorithms GOA
global optimisation algorithms	
HJ	hookes-jeeves pattern search
LOA	local optimisation algorithms
MIP	mixed integer programming
NSGA-II	non-dominated sorting genetic algorithm
PSO	particle swarm optimisation
SA	simulated annealing

Contents

List of Figures

List of Tables

1
Introduction

This chapter presents the background to the research topic, in addition to purpose, research questions, delimitation, background of AFRY and ethical considerations.

1.1 Background

Simulation tools, such as discrete-event simulation (DES), have been used to model, simulate, and analyse systems since the 1950s [1]. DES provides a representation of a real-world system, by simulating the dynamics on an event-by-event basis, for which it generates a performance report [2]. This method of simulation focuses on analysing the duration that entities spend navigating through a series of specific events within a system, and is commonly used for logistic modelling and queuing problems [3]. DES is one of the most powerful tools for designing, planning, and improving material flows [4], and is ranked as one of the best decision support tools [5].

Optimisation models are mathematical representations of decision-making challenges, offering solutions that aim to identify the most optimal decision based on the defined objective function and the specific constraints [6]. These models serve as valuable aids in information systems, assisting decision-makers, and also play a crucial role as analytical instruments for enhancing the design of information systems [7].

Various complex real-world systems, e.g production systems, can effectively be modelled using discrete-event simulations. Simulation using DES typically involves finding the best configurations and utilisation of resources such as machinery, buffers, and scheduling [2]. DES models can, to some extent, be used to optimise processes by experimenting with various parameter inputs. However, as the number of potential solutions grows, it becomes impractical to test all combinations. In such cases, optimisation methods can offer a more effective solution. By optimising simulation models, companies can identify improvements for a factory and supply chain before committing to an investment.

Discrete-event simulation can be used as a test bed for testing potential solutions for production systems or logistic systems. The simulation model outputs a simulated result answering how the simulated scenario would play out [1]. In an optimisation model, the user specifies all parameters and conditions and returns the most optimal solution. An example of a possible combination of these two models could let the user build a DES model and an optimisation model with all parameters and conditions.

The optimisation model could output solutions for the simulation model, which runs until it encounters a decision that can be supported by optimisation, at which point the simulation model could output all available information which is sent back to the optimisation model. The optimisation model could take the new information from the simulation model and creates new solutions for the simulation model and the process is repeated until the optimal solution is found [8].

1.2 AFRY

AFRY, formerly known as ÅF & Pöyry, is a global company with offices in over 40 countries and over 19,000 employees. The company holds a leading position as a market leader within the European engineering, advisory services, and design sector. In 2021, the company's net sales were approximately 24 billion SEK [9]. The major divisions of AFRY, which are Energy, Industrial & Digital Solutions, Infrastructure, Management Consulting, and Process Industries cover a broad range of industries which makes them a constant presence in engineering projects [9]. AFRY has expressed interest in investigating potential areas where simulation optimisation is of use, and potentially develop methods for optimising a simulation model. Due to time constraints and other commitments, AFRY cannot justify assigning full-time engineers to such a project. This thesis will be conducted mainly on-site at their largest office in Gothenburg within the Supply Chain Management department.

1.3 Purpose

The purpose of this thesis is to further explore and investigate the application of a combination of simulation and optimisation techniques in the context of discrete-event simulation models for various complex real-world systems. The primary objective is to identify potential areas where a combination of simulation and optimisation can be effectively employed to support strategic business decisions within production and intralogistic systems. By combining optimisation techniques with simulation models, the authors aim to provide a step forward in the field of DES.

1.4 Research gaps

While it was found during literature study that there have been implementation of optimisation in discrete-event simulation, and the subject at hand has been studied, the authors believe that their is a shortage of decision support tools that assist the users in the implementation of simulation optimisation. The authors also find a research gap in seeing where and when it is useful to use simulation optimisation in industry. The aim is to further bridge this gap and provide insight for industries that search to venture into simulation optimisation.

1.5 Research questions

The following are the stated research questions for this thesis.

- (RQ 1) What optimisation methods are used in DES and what are the potential impacts of combining them with DES?
- (RQ 2) What are the criteria for a DES case to qualify for optimisation?

1.6 Delimitations

DES is the only simulation method that is considered in this thesis due to the fact that the simulation team at AFRY Gothenburg primarily works with discrete-event simulation and that it is widely used across many industries. Restricting the thesis to only DES is beneficial because of the extensive data library and competence that AFRY can offer. This thesis will focus on the simulation of manufacturing systems and intralogistics systems due to it being the main focus of the simulation team at AFRY Gothenburg.

1.7 Ethical considerations

During all interviews, both internal and external stakeholders, the authors have been thorough to ask each interviewee if they consent with being recorded for transcribing purposes. The transcriptions are solely for data analysis and all interviewees are to remain anonymous throughout the entire thesis. The authors have been careful with wording and formulating questions in a non-leading way in order to stay unbiased in the qualitative portion of the thesis. The authors possess prior knowledge in the subject of discrete-event simulation gained during their studies, this knowledge was beneficial when understanding certain subjects but that did not affect the results nor the direction of this thesis.

2
Theory

This chapter presents information regarding DES and optimisation methods with relevant literature and theories to establish a foundation and understanding of the different topics of this thesis. This chapter forms the theoretical basis of this thesis.

2.1 Discrete-event simulation

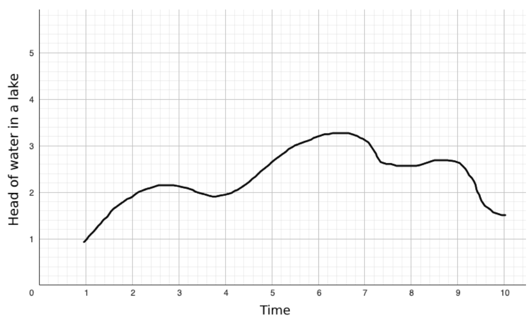
Discrete-event simulation serves as a valuable tool for modelling various systems, with a primary focus on queuing systems. In this modelling approach, a system is depicted as entities progressing from one activity, effectively represented as a time delay, to another activity, with queues separating these activities. Queues arise when entities arrive at a pace faster than they can be processed in the subsequent activity [1]. The concept of DES can be applied to a broad range of systems, whether involving people, physical items or information, it can be conceptualised as queuing systems, with entities moving through the system [10]. As a result of this, DES finds widespread utility across a multitude of organisations and industries [11].

In contexts such as production and supply chain management, discrete-event simulation plays a crucial role in optimising various aspects. It helps enhance production processes, manage inventory effectively, and streamline logistics by simulating workflows, production lines, and distribution networks [11]. The primary goal is to identify bottlenecks, enhance resource utilisation, and minimise delays, ultimately leading to more efficient operations [2].

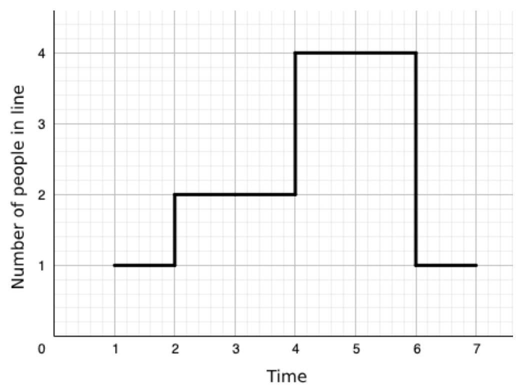
According to [11], a typical simulation using discrete-event simulation looks as follows; a simulation clock stores and presents the time spent in simulation, this clock moves forward in discrete intervals of varying sizes throughout the run. At a discrete event within the simulation, the clock halts and all the actions within the model is taken at this specific point in time, when this is done the simulation clock advances to the next discrete event where the applicable actions are taken. A run of a DES model can thus be split into two major parts within a loop, "simulate all actions required at the current simulated time", and "proceed the simulated clock to the next required action" [11].

The stochastic nature of a DES model is both beneficial and disadvantageous, since the results of a simulation can vary substantially from one run to the other. This creates the need to run the simulation numerous times in order to achieve an understanding of an accurate result. The variations in results when simulating the same model can be referred to as stochastic noise. While this uses more time and computing power it also corresponds to a real-world scenario, where outputs and efficiencies cannot be guaranteed and always contains some level of uncertainty [12].

It is vital to make the distinction between a continuous system and a system that can be considered to be discrete. While no real-world system can be accurately labelled as solely continuous or discrete it is still possible to separate the two systems, this is due to a majority of systems having tendencies towards one of the two [13]. A continuous system is a system where the state variable(s) fluctuate continuously over time, a common example of this is the water head of a lake. During a period of rain the water level of the lake raises continuously and through evaporation and outlets the water levels will continuously decrease to the normal level again. See Figure 2.1a for a visual representation. A discrete system, on the other hand, is a system where the variable(s) only changes at discrete events in time, which a queuing system represents. An example of this may be a number of people are waiting in line to a restaurant. The number of people in lines changes when someone enters or exits the line whilst the variable remains unchanged during the time in between [14], see Figure 2.1b for a visual representation. With this distinction made, the systems studied in this thesis will be discrete systems.



(a) Example of a continuous system



(b) Example of a discrete system

Figure 2.1: Examples of continuous and discrete systems [14]

2.2 Optimisation methods

There are different optimisation methods that use optimisation models in order to provide a systematic approach to tackling decision problems by representing them mathematically. They are used across various domains to find the best solution from a set of given parameters. These models typically involve defining an objective function to be maximised or minimised, along with constraints that must be satisfied [7]. Optimisation techniques, such as linear programming, integer programming, and dynamic programming, are applied to solve these models and identify optimal decisions. The information provided by optimisation models can inform strategic planning, resource allocation, and process improvement efforts within organisation, making them invaluable tools for decision support and system design [15].

Utilising optimisation is resource demanding, no matter which method is used. Less complex systems with fewer variables and parameters require less computing power. As soon as a system becomes more complicated with more parameters and variables, the complexity increases rapidly along with the difficulty, demand of computing power, and cost of the optimisation [16]. In cases where real-world systems get substantially complex, utilising optimisation methods could be too resource demanding and yield an increase in cost that is not justifiable [16][17].

Optimisation methods can be split into two groups, global optimisation algorithms (GOA) and local optimisation algorithms (LOA). A system optimised using LOA tends to converge towards the first optima that is encountered, this could be a local optima resulting in the algorithm being satisfied with a solution that might not be the optimal. LOA tends to have difficulties solving discrete optimisation problems [16]. Discrete optimisation is decision making based on integers or binary parameters, where there are distinct alternatives. An example of this may be to do a task, or to not do a task, alternatively to turn right, or not to turn right [18]. While a system optimised with GOA cannot guarantee that a global optima is found, it is less inclined to converge towards a local optima. Global optimisation algorithms are considered to be more powerful than local optimisation algorithms since LOA experiences issues when solving optimisation problems with integer and/or discrete variables. Finding a local optima is not necessarily a bad result and is still considered as optimisation of a system but GOA provides a larger chance of finding a more optimal result [16].

2.2.1 Stochastic algorithms

Using stochastic algorithms is a common technique to solve global optimisation problems due to these algorithms being flexible and robust [19]. Stochastic algorithms cover optimisation methodologies that incorporate the generation and utilisation of random variables. Within the domain of stochastic problems, these random variables are integral components of the optimisation problem formulation, manifesting as either random objective functions or random constraints. Additionally, stochastic algorithms comprise techniques where random iterates are employed [20].

Evolutionary Algorithms (EA), a type of stochastic algorithm, have a background in biological evolution based on Charles Darwin's theory around 'survival of the fittest' [21]. The shared foundation among evolutionary algorithms lies in the premise that within a population inhabiting an environment characterised by finite resources, competition for these resources take place, thereby precipitating the mechanism of natural selection favouring individuals best adapted for their surroundings, i.e. survival of the fittest [22].

EA are conceptually rooted in biological evolution, drawing inspiration from mechanisms as reproduction, mutation, recombination, and selection. Initially, a randomised set of candidate solutions, representing elements within the function domain, is generated with the objective of maximising a quality function [21]. Subsequently, this quality function, typically formulated as an abstract fitness function, is applied within the problem domain. Through a process similar to natural selection, superior candidates are identified based on their fitness, utilising recombination and/or mutation techniques. Recombination, facilitated by binary operators, involves the fusion of genetic material from two or more selected candidates, or parents, to produce one or more new candidates, or children. Mutations instead operates on a single candidate, generating a singular offspring. Following a recombination or mutation, a new set of candidates is formed, guided by their fitness values. This iterative procedure persists until a sufficiently optimal candidate is obtained [19].

An additional popular stochastic algorithm is the simulated annealing algorithm (SA). SA, similar to the evolutionary algorithm, has its roots in nature and is based on the theory behind the annealing process of ideal crystals in thermodynamics [23]. The annealing process of metal is a stochastic activity due to the chaotic behaviour of molecular movements caused by swift or prolonged cooling [24]. The algorithm achieves this stochasticity through the acceptance of inferior solutions throughout the process. This acceptance enables the algorithm to break away from a local optima [25]. SA was the first algorithm method to adopt this type of behaviour of allowing inferior solutions and paved the road for other similar algorithms such as tabu search [23].

2.2.2 Deterministic algorithms

Deterministic methods leverage the inherent analytical characteristics of a problem to iteratively generate a sequence of points, aiming to converge to wards a global optima. While heuristic methods offer greater flexibility and efficiency compared to the deterministic approaches, they do not ensure the quality of the solution obtained. Furthermore, as the complexity of the problem increases, the likelihood of identifying the global solution diminishes. Deterministic techniques such as linear programming and non-linear programming offer versatile tools for addressing optimisation problems, aiding in achieving a global or approximate global optima [26].

Non-Linear programming is considered the "jack of all trades" of optimisation models [27]. Many problems that arise in the world does not necessarily have linear solutions, twice as many workers does not always result in twice the efficiency [27]. Non-linear programming requires a lot of computing power and is often very hard to solve compared to its linear counterpart [28]. Non-linear programming suffers from the same dilemma brought up with local optimisations methods where, if a solution is found it could be a local optima or even a stationary point [29]. Most non-linear programming is based around Newton's method by approximating the non-linear problem linearly to get an estimation of an optima. This process is repeated until a satisfied result is found [28].

Linear programming was first brought around in the late 1940s [30] shortly after the second world war. Along with the fast development of computers since this time this technique has rapidly evolved. Linear programming offers the capability to state general objectives and devise a strategic road map for making the optimal decision when faced with real-world scenarios. It is able to formulate complex problems into precise mathematical models, use algorithms to solve these models and utilise computational resources in order to obtain a set goal in the best, or close to best, possible way [31]. Linear programming is closely related to linear algebra with one of the main differences being that linear programming primarily uses inequalities rather than equalities that is often used in linear algebra [32].

Linear programming operates through a systematic mathematical framework designed to optimise a given objective function to a set of linear constraints. The process of solving a linear programming problem typically starts with the formulation of said objective function, which represents the quantity to be optimised [32]. The constraints are established to reflect the limitations or requirements of the problem and are expressed in terms of the decision variables, i.e. the variables that represent its contribution to the objective. Once the objective function and constraints are defined, the next step involves the utilisation of a optimisation algorithm to identify the optimal solution to the problem [33]. These algorithms systematically explore the feasible region defined by the constraints to connect the optimal values of the decision variables that either maximise or minimise the objective function while satisfying all constraints. Figure 2.2 shows an example of how a feasible region may look. The coloured area is the region that is defined by the constraints, and the intersections between lines represent possible solutions to the problem [34].

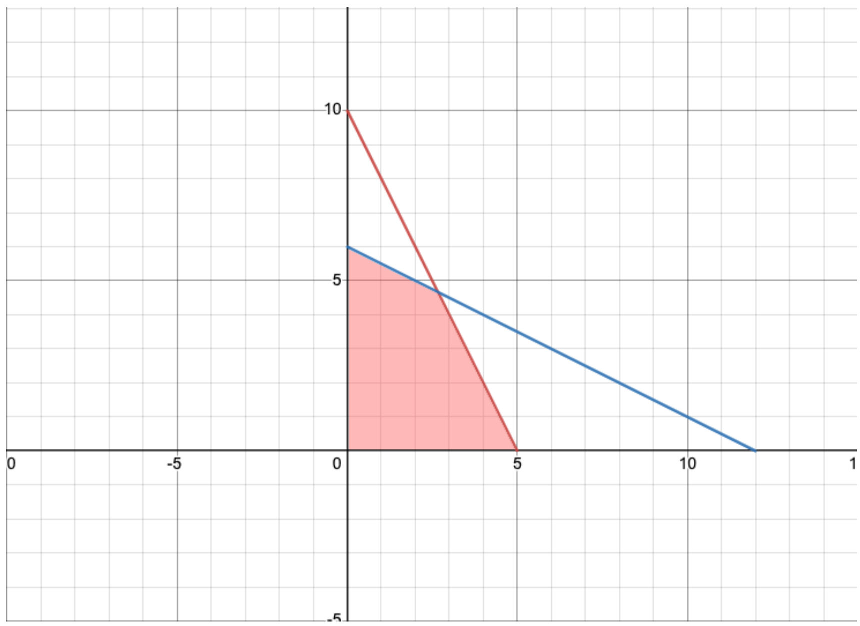


Figure 2.2: Visualisation of an example of a linear programming problem

Linear programming is useful when working with continuous problems and solutions, however, when working with discrete variables one can utilise integer programming instead. In certain contexts, it is necessary for variables to be discrete or integer [35]. Such contexts may arise in engineering problems where a design variable is discrete if its value must be chosen from a finite set of predetermined values, such as standardised measurements. In some cases, the variables needs to be integers, such as the amount of bolts needed in an assembly, or capacity in a buffer, these can be solved with integer programming [36]. Figure 2.3 illustrates an integer programming problem with the possible solutions marked with green circles. The number of possible solutions drastically decreases in comparison to a similar linear programming problem, this does however not mean that integer programming problems are easier or faster to solve. This is due to the fact that integer programming adds new constraints where the result is required to be integer or discrete, leading to the solution demanding more computing power and time [35][37].

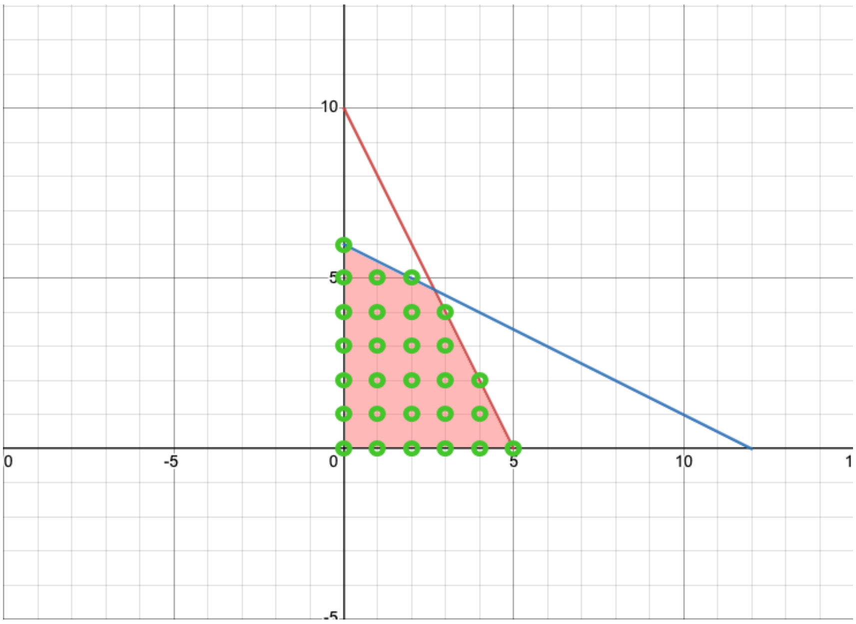


Figure 2.3: Visualisation of an example of an integer programming problem

2.3 Simulation optimisation

Throughout at least the last two decades the combination of simulation and optimisation has been a re-occurring subject during the Winter Simulation Conference[38] and a numerous of different articles and researchers share their opinions, thoughts and progress in simulation optimisation. These studies and articles presents a varying degree of theoretical and practical implementation. Brief summaries of how some of these implementations were made will be provided in this section.

Simulation optimisation was tested in a real-world scenario and studied the Viennese subway system. The aim of the study was to optimise the time interval between trains for the subway network, optimising on the minimising of costs as well as minimising the average wait time for each passenger [39]. This study used a discrete-event simulation approach with the use of various evolutionary algorithms, of which ”covariance matrix adaptation evolution strategy Covariance Matrix Adaptation Evolution Strategy (CMA-ES) performed the best and provided a demonstration of the practical applicability of evolutionary algorithm approaches [39].

Another example of where optimisation was implemented in a simulation was when simulating a real-world scenario in the aerospace industry. In this case, the main objective was to ramp-up profiles of the assembly lines in order to secure on-time deliveries, balance requirements of the workforce, and to minimise buffer times [40]. To address the challenge at hand, the authors of the paper developed an approach of the hybrid variant, that integrated DES in FACTS Analyzer with mixed-integer linear programming and multi-objective optimisation. In this case, a Non-Dominated Sorting Genetic Algorithm (NSGA-II) algorithm was used in the optimisation process [40]. The simulation model created included parameters such as buffer usage, lead times, production rates, and workforce requirements that were used to generate reports that would support the decision-making process at later stages [40]. The model that was created in this project succeeded to minimise the time that the assemblies waited in buffers, which led to well-managed assembly lines and timely deliveries.

The combination of simulation and optimisation was also tried in a case of an automotive manufacturing workshop. In this instance, the main focus of the project was to optimise the production logistics using the DES software FlexSim with an unspecified optimisation algorithm. This study used Petri net modelling [41] to analyse the discrete-event system created in FlexSim. The authors in this study describe how they aimed to set a rational tact time for which the production line should operate, where the goals included minimising idle times, avoiding bottlenecks, and balancing workloads across work stations [42]. According to the authors, the impact of using optimisation in this project was significant, as the tact time was able to be reduced from three and a half minutes to one minute and forty-five seconds. The authors also stated that the workload of each station became more balanced, leading to a more efficient production flow [42].

There has also been applications of simulation optimisation in iron production, where a company needs to maintain a large amount of inventory of conveyors in the case of needing to make prompt replacements. Maintaining this large stock of conveyors can be costly and thus the implementation of simulation optimisation was tested in order to minimise the required inventory [43]. With the help of historical data a simulation model was created to replicate the failures over a long period of time, Mixed Integer Programming (MIP) was then formulated and developed to optimise the minimisation of downtime [43]. By leveraging the verified historical data, simulation modelling and optimisation algorithms this implementation of simulation optimisation managed to optimise inventory profile and decrease of cost, find a risk-reward balance and provide a decision support tool for the future of the company [43].

In an article by Henderson and Jing from 2015 [44], a case is discussed regarding optimising a bike-sharing system in New York City with the help of simulation optimisation. The primary goal was to determine how to position bikes across the various stations in preparations for the increased demand during mornings, in order to minimise any customer inconvenience. The authors describe how the arrival and departure of bikes were modelled using Poisson processes [45]. In this case, the optimisation algorithms of choice was Sample-Average Approximation and Random Search, where Sample-Average Approximation use a set sample size to estimate the objective function for the given problem and applies an unspecified deterministic optimisation algorithm to minimise the sample-average function [44][46]. The Random Search methods iteratively evaluate any potential solutions and select the best performing ones. The authors claim that the impact of applying these optimisation methods is of importance and that in this case, starting with intuitive solutions such as half-filling stations and refining those solutions using optimisation tools led to the optimal allocation of resources [44].

Discrete-event simulation is often used in the healthcare industry and there have been implementations made of simulation optimisation as well. One article from the Winter Simulation Conference in 2015 written by Tarun Mohan Lal, et al [47]. discusses how simulation optimisation can address complex decision-making problems in the field of healthcare. The rising costs and need for efficiency in the healthcare industry has created a high demand for new decision-support tools, in this article MIP is implemented into a DES model developed based on real hospital data in order to optimise personnel schedules and managed to minimise the previous errors between patient load and scheduled personnel [47]. This study managed to successfully integrate simulation optimisation into the healthcare industry based on verified data, paving the way for similar project to take place.

Looking past discrete-event simulation, there has been practical implementations of simulation optimisation through the use of EPANET hydraulic simulation toolkit targeting pump operations within the distribution network of water [48]. This study implemented a multi-threaded simulation optimisation platform created using C++ that employed three different types of optimisation algorithms, Particle Swarm Optimisation (PSO), Hookes-Jeeves Pattern Search (HJ), and Genetic Algorithm (GA) [48]. This platform was tested in a real-world scenario in a portable water distribution system in southern California, where the optimisation methods were compared in efficiency of reducing water pump energy usage [48]. In this scenario, with the EPANET hydraulic simulation toolkit, PSO was found to be the most appropriate and effective optimisation approach [48].

In a more theoretical setting, the combination of DES and optimisation has been tested on randomly generated instances of the Server Allocation Problem [49]. In this article, the problem is in regards to determining the optimal number of parallel servers to allocate at every stage of a system in order to make sure that the average time customers spend in the system is below a set target threshold, while still managing to minimise the overall server cost [49]. The authors describe how the selected optimisation algorithm was a simulation-based Benders Decomposition method, which is an adaptation of the traditional Benders Decomposition approach [50]. In the method used, instead of solving mathematical programming problems to generate cuts like in the traditional method, the cuts are derived using information gotten directly from the sample path of the simulation. The results presented in the article demonstrated that the effectiveness of the approach used reduced computational requirements and improved the solution quality [49].

Despite what was found through the Winter Simulation Conference database, the authors believe that there are still things missing. The two major research gaps, presented in Section 1.4, remain to be studied in this thesis.

3
Methods

The methodology used in this thesis was a case study [51]. The case study includes a triangulation of methods such as a literature study, and a qualitative study. The qualitative study included interviews with internal, as well as external stakeholders. Secondary data was also collected as part of the qualitative study. This was done to gather information from various sources and for the ability to verify and complement data from literature with data from the qualitative study and vice versa. More specific information about each of these is found in their respective sections below.

3.1 Literature study

The literature study in this thesis aims to examine the current state of knowledge regarding DES models, optimisation techniques, and whether the combination of the two is feasible in order to potentially find solutions to queries that a simulation engineer might not have detected. The objective of the study is to provide a comprehensive review of the available literature on the feasibility, performance, and impact on efficiency on discrete-event simulation. In order to find the best suited literature for this project, a combination of relatively new and old literature will be used. The older literature is relevant since the concept of discrete-event simulation has largely remained the same since the 1990s with recent changes mainly revolving around visual improvements and calculation speeds [1][52]. These forms of changes do not impact the foundation of how DES functions and thus does not affect this thesis. The scientific databases used for gathering literature are listed below.

- Scopus
- Web of Science
- Google Scholar
- ScienceDirect

To find the most relevant publications for this thesis, the following keywords were used across all databases listed above: discrete-event simulation, DES, simulation, optimisation, simulation optimisation, simulation methods, optimisation methods, production systems, logistic systems, logistic modelling, simulation techniques, optimisation techniques, DES optimisation. To determine whether the articles were relevant to this thesis, the authors read the abstract, introduction, and conclusion of the information source.

To get further understanding in when optimisation and simulation has been used in combination and conjunction, a complementary literature study was conducted that focused on the article database from the winter simulation conference. Similar keywords were used from the previous literature study and the same strategy for finding relevant articles was used. This literature study only considered articles dated from 2004 and onwards.

3.2 Qualitative study

A qualitative study will be conducted in order to gather useful data from both internal and external stakeholders within the studied field. Internal stakeholders refers to people working at AFRY Gothenburg and external stakeholders refers to experts in the DES field or academics studying the subject. The aim of this study is to ask questions to people working or researching in this field to learn more about the many techniques, methods, and opportunities that are available in order to obtain a comprehensive understanding of the subject at hand as well as help answer the research questions stated in Section 1.5. The main focus of the qualitative study is to act as a complement to the literature study, but can also be used to provide information that cannot be found in the literature study.

The interviews with the internal stakeholders will be separate from the interviews with the external stakeholders. The internal stakeholders are directly impacted of the results of this thesis. These interviews are held in order to get a broader picture of the current situation at AFRY regarding optimisation using DES and their current processes, methods, and tools. The external stakeholders involve people working for developers of DES softwares, academics within the field of DES and optimisation, and other people with experience and knowledge within simulation. For the interviews with the internal stakeholders, the questions were designed to get a deeper understanding of the current level of knowledge within DES and optimisation at AFRY. These questions can be seen in Appendix A. Interviews with the external stakeholders can provide information about, but not limited to, available optimisation tools within DES softwares, current research and state of the art of optimisation tools and other companies' knowledge in this field. These external stakeholders have experience in, and frequently use Plant Simulation, AutoMod, and Facts Analyzer as their DES software of choice and all have extensive knowledge within optimisation in DES. The questions for the interviews with external stakeholders can be seen in Appendix B.

Tables 3.1 and 3.2 presents brief profiles of the interviewees including their role and their experience with DES. Table 3.1 presents the profiles for the five internal interviewees and Table 3.2 presents the profiles of the three external interviewees.

Role	Experience within DES
Simulation specialist	< 1 year
Simulation specialist	5.5 years
Consultant in simulation and production development	1 year
Simulation specialist	2 years
Team leader & simulation specialist	5 years

Table 3.1: Internal interview profiles

Role	Experience within DES
Implementation consultant	5 years
Research and development engineer	7 years
Senior process engineer	> 30 years

Table 3.2: External interview profiles

3.2.1 Data collection

This section describes some of the data collection during the qualitative study, which includes the structured interviews form the internal stakeholder interviews and external stakeholder interviews as well as the unstructured interviews in forms of discussions and conversations with employees of AFRY Gothenburg regarding prior projects.

3.2.1.1 Structured interviews

To get an understanding why some of the questions was asked, the following section aims to provide further explanation to some of the questions' importance. This does not include all the questions and is merely an extract from the full list of questions found in Appendix A and B.

The question "*What proportion of your projects are improvement/optimisation projects, e.g. improving the output of a factory?*" was asked during the internal interviews to achieve an understanding of the extent of projects that optimisation tools may be suitable for. It is the authors' understanding that utilising optimisation tools may be more suitable on certain types of projects compared to others, e.g. increasing the throughput in a factory compared to validating a production layout proposal.

"*How do you arrive at potential solutions/improvements in your work?*" was asked to the internal stakeholders to get an idea of the workflow that the interviewee normally follows in their work process. This may help give an understanding to whether the utilisation of optimisation tools would be a significant change in their workflow.

"*What would it take for you/your team to start using optimisation in conjunction with your simulation models?*" was asked to the internal stakeholders to gain insight of what is expected from the stakeholder regarding the use of optimisation tools in their work process.

The external stakeholders was asked, amongst other things, "*In what cases do you consider it appropriate to use optimisation tools?*" to identify the type(s) of projects and/or queries where they believe it to be beneficial to utilise optimisation tools. The answers to this question might strengthen the responses to the related question, "*What proportion of your projects are improvement/optimisation projects, e.g. improving the output of a factory?*", that will be asked to the internal stakeholders.

"*Does the simulation engineer need to provide anything additional when building the model to use the optimisation tool? Do you need to consider optimisation when building the simulation model?*" was asked to the external stakeholders in order to comprehend whether the current workflow and processes of the simulation engineer is subject to change when implementing the use of optimisation tools.

In order to gather a broad range of perspectives and insights relevant to the research questions stated in Section 1.5, interviews were held with individuals with expertise and experience relevant to the study. The interviewees selected represent a range of backgrounds, including people from the industry, academics and technology providers of DES software. The diverse perspective and experience can provide insights that will strengthen the understanding of the subject.

3.2.1.2 Unstructured interviews

Throughout the entirety of the project the authors collected data in the form of unstructured interviews with employees in the simulation team at AFRY Gothenburg and access to data from prior simulation projects. The constant conversations and data flow allowed the authors to get a deeper understanding in the field of DES and optimisation, constant coaching and help from experts in these fields, as well as insight into how project queries are commonly formulated.

The majority of the data collected from the this was aimed to become a foundation for the decision tree that was later created. The data was collected through conversations with project leads for the respective project, the project leads presented questions they asked themselves prior to starting said project, challenges with the project and their own thoughts regarding whether or not they believe that optimisation tools could have benefited them. The data collected was gathered in a larger document containing several different projects, ranging in size and goal. The document was then summarised and the questions the simulation engineer ask themselves prior to starting a simulation project along with the challenges and outcomes of the project then formed a basis of questions for the majority of the nodes on the decision tree.

3.2.2 Data analysis

Each interview was recorded with the permission of the interviewee in order to ease the transcription process. After each interview the transcription process was started immediately when the memory of the conversations was still fresh, AI transcription tools were also used in order to speed up the process of transcribing [53][54], this was however done under close supervision by the authors to ensure that no information was lost or incorrect.

Oce all of the internal stakeholder interviews and transcription were completed, all of the transcribed data from each interview was collected and placed in a large combined document. Each question was listed and the answers from all of the interviewees were sorted under the corresponding question. The different interview answers were colour-coded in order to make sure they would not be mixed up.

Once this document was completed the data analysis process could begin, the authors scanned through each question and searched for the identification of similarities or differences in the answers between the different interviews and this was then summarised at the end of each question. It was important to ensure that no information was lost from the interviews and that each answer and viewpoint was taken into account, in cases where the answers were divergent, there would instead be multiple sets of summaries reflecting the different perspectives. This was then repeated for all of the questions for the internal stakeholder interviews. This process was then replicated for all of the external stakeholder interviews.

The summaries were then reviewed further and colour-coded once again for the different perspectives and point of views, this made it easier to look back to when comparing the data from the internal stakeholder interviews with data from the external stakeholder interviews, but it also facilitated the process of comparing and complementing the literature study with the qualitative study.

4
Results

The following chapter presents the results from the literature study and qualitative study in sections of optimisation tools in DES software, the choice of optimisation algorithm and finding the right use-case.

4.1 Optimisation tools in DES software

Results from the literature study showed that optimisation tools already exist in multiple DES software including but not limited to, Plant Simulation, AutoMod and FlexSim [55][56][57]. All of the mentioned software is currently in use by the simulation team at AFRY Gothenburg. Beyond the built-in optimisation tools the literature study also proved that it is possible to create customised optimisation tools in external programs and bridge it to a DES model. This provides the user much larger degrees of freedom in the sense of altering the algorithms in comparison to the built-in tools offered in DES software. Although the user gains customisability and more control over the algorithm and its functions, it is shown to be a much more time consuming task compared to using the tools offered by the DES software developers, along with a heightened required level of knowledge.

Literature pointed towards the use of several different algorithm methods in implemented optimisation tools in DES software, such as random search, simulated annealing, tabu search, and evolutionary algorithms. The articles found in the literature study comparing these algorithms all showed that evolutionary algorithms and simulated annealing are the most suitable optimisation method for the DES software studied further [58]. This was further strengthened by statements from the external interviews conducted in the qualitative study.

Results from the qualitative study showed that the external stakeholders utilise the built-in optimisation tools in their respective software with varying degree of regularity. This shows that use-cases for optimisation can be found. External interviews also showed that even though the use-cases for optimisation do exist, the majority of projects are verification or validation projects where the client wishes to ensure that their planned production functions as intended. This was in the internal interviews, confirmed that a large portion of their projects were also verification or validation of planned production systems.

During the interviews with the internal stakeholders, it was found that none of them have used or tried optimisation tools in conjunction with DES, whether those were built-in tools or external, customised tools. Prior to this thesis, a majority of the internal stakeholders were not aware that optimisation tools exist within the DES software that they use. One interviewee had tried it during their studies, but would not consider themselves to have any significant knowledge in the area.

According to the interviews with the external stakeholders, the difficulty of implementing optimisation tools in a model depends on the DES software that is to be used. From the interviews it is evident that AutoMod demands the user to keep the optimisation in mind from the start when building the simulation model. Instead, Plant Simulation is more plug and play where the user does not have to keep optimisation in mind when building the model. According to the interview with the corresponding external stakeholder, this also results in the user being able to use the built-in optimisation tools on older models with little or no modifications needed when using Plant Simulation. The qualitative study also showed that the same is true about Facts Analyzer, according to an interviewee with expertise in this software, it is not necessary to keep optimisation in mind when the simulation engineer first builds the DES model. No further information was found about FlexSim regarding ease of use and potential preparations needed in order to utilise optimisation tools.

One of the external stakeholders mentioned during their interview that Siemens has a web-design software called Mendix, which allows the user to design the experiments in the design tool and then run the trials server-side. Since the experimental runs would not happen client-side, this software could greatly reduce the required computing power, which can otherwise be a limiting factor in scenarios such as these. The interviewee also stated opportunities of using artificial intelligence in the software. Mendix is a low-code platform, meaning it required little to no coding knowledge which would lower the requirements on the user. Implementation of this could allow purchasers of simulation models to run experiments and make modifications to their system based on results from the platform.

4.2 The choice of optimisation algorithm

One of the main reasons as to why optimisation using EA scored so high in these studies is due to its ability to escape local optima where other methods tend to get stuck. Simulated annealing is another algorithm that is able to break loose from local optima solutions and scores high in studies. The benefit of being able to escape local optima is important in optimisation tools for DES, since the algorithm can otherwise linger at solutions which the algorithm believes is the global optima. Although evolutionary algorithms and simulated annealing specialises on not getting stuck on local optima, they do not necessarily ensure that it reaches the global optima either. They are however more likely approach the global optima in a more efficient way than the other tested algorithms and methods in articles and cases found in the literature study. They are both able to do this efficiently and with relatively low computing power compared to other tested algorithms. In a majority of the cases, discovered in the literature study, the fact that these optimisation algorithms does not guarantee a global optima localisation it can still provide a solution that is considered to optimise the tested DES models.

Even though EA and SA, two global optimisation algorithms, perform well in this article [59], the literature study found that, for no specific reason, it is more common to use EA over SA when it comes to optimisation within DES [59]. SA, in several studies [12][58], proved to be very efficient in the few cases where it was applied into DES. One article in particular showed critique towards the fact that a majority of studies showed favouritism towards EA over the similarly performing SA method [59].

Interestingly, Plant Simulation, AutoMod, and FlexSim all use evolutionary algorithms in their built-in optimisation tools which reinforces the applicability of using these types of optimisation methods within DES. Further, when exploring customised optimisation tools it was found that EA was an equally popular method in these scenarios [60]. As stated above, the customised optimisation tools provide the user with a larger degree of freedom and can also help innovate and discover new ways of improving optimisation within DES. Hybridisation of EA and SA is discussed briefly in the same article [59] that criticised the lack of usage of SA in integrated optimisation tools within DES. The article explains that a hybrid version of EA and SA utilise evolutionary algorithms to generate preliminary process plans and simulated annealing is then used to find alternative optimal or near optimal process plans. This type of hybridisation was proven to be both more robust and result in better optimisation results compared to using the two algorithms separately [59].

According to all of the external stakeholders, the built-in optimisation tools are rather generic and tends to be adequate in most cases that would benefit from optimisation. When the simulation engineer instead decides on bridging a customised optimisation algorithm with a DES software for more specific types of problems their shared opinion was that it will almost always outperform the built-in tools, but it requires a great amount of extra time spent on both optimising as well as preparatory work on creating the optimisation tool. This was further strengthened by an article from the literature study that compared efficiency and result of built-in tools with customised optimisation algorithms [60].

When asked what the reasoning may be behind all of the above mentioned DES software using evolutionary algorithms, the external stakeholders responded similarly. One of the interviewees stated that "*Evolutionary algorithms are rather easy solutions in order to simulate several experiments, it works extra well when optimising something that has a large number of different parameters*".

"*Evolutionary and genetic algorithms functions really well in the type of problems that was faced during my research, these algorithms were the best performing when solving these types of problems*" was answered by another interviewee when asked the same question. The interviewee with the most experience in the area did not necessarily know the reasoning behind the choice of using evolutionary algorithms in lieu of other alternatives. One external stakeholder mentioned during their interview that they have not, neither in their academic work nor professional career, compared or investigated the differences and/or similarities between evolutionary algorithms and other optimisation algorithms.

4.3 Finding the right use-case

The external stakeholders described during their interviews that one of the most important steps in the optimisation process is to find the right use-case. Not all problem types that tends to be solved with DES is necessarily something that can benefit from the use of optimisation tools. Regarding the use of optimisation systems, an external stakeholder stated that "*Using optimisation tools is appropriate when one aims to make improvements in their system, such as increase their production, reach KPIs, reduce set-up times, or decrease lead-times*". Similar thoughts were expressed during the interviews with the internal stakeholders, where the interviewees speculated that utilising optimisation tools may be suitable for these kinds of projects.

Through the qualitative study an external stakeholder stated that the most common request from their customers is validation- and verification-type projects, where the client has a prototype or an already planned system. The task for the simulation engineer is to build a DES model to test and verify the concept from the customer. In these cases there are usually not a lot of possibilities and time available to optimise and change the systems foundations, nor is this the purpose of such projects.

Comparing this to the results from the internal stakeholder interviews it can be seen that this is also the case for a large portion of projects at AFRY. "*There are a lot of concept evaluation projects, where the customers wish to test and validate their current system*" and "*Normally the case is that the client is looking to verify certain scenarios, i.e if the planned capacity is achievable with given parameters*". As mentioned previously, these kinds of projects are not suitable nor would they benefit from the implementation of optimisation tools, whether they are part of the DES software or external, customised algorithms. In the experience of the internal stakeholders, the client is oftentimes aware of their bottlenecks and their locations and because of this they are more interested in finding answers to capacity questions. In the projects where the internal stakeholders do consider it appropriate to use optimisation tools they are often constrained by time and have other priorities.

The proportion of projects where optimisation tools may be feasible differs among the internal stakeholders. According to all but one of the internal stakeholders, a very small portion of projects have queries and enough time that would allow them to utilise optimisation tools to its full extent. However, one of the internal interviewees stated that a large portion of their projects include room for the engineer to give their input on improvement possibilities, these projects could possibly benefit from the implementation of optimisation.

During the interviews with the internal stakeholders, several interviewees mentioned greenfield and brownfield projects. A greenfield project is when a new system is being built and a brownfield project is a project on land that is already in use, where there might be an existing system that is being modified. The interviewees stated that during a greenfield project the client oftentimes have a plan for what should be done and the layout and wish to verify that it will work as intended. In greenfield projects the data is often limited and based on estimations which could hinder the usage of optimisations due to the uncertainty of the results from the data that is provided. This leads to greenfield projects often being classified as verification- and validation projects. In a brownfield project, the client, and in turn the simulation engineer, typically have access to a larger amount and more accurate data. This allows for more precise models and predictions which is beneficial when using optimisation tools. According to the internal interviews, greenfield projects are currently more frequent than brownfield projects which might be a contributing factor in these stakeholders not having tried the available optimisation tools.

4.4 Creation of decision tree

This section aims to explain how the results from mainly the qualitative study and the secondary data collected helped formulate the components that make up the decision tree. Each of the following subsections explain the origin of each corresponding node in the decision tree.

4.4.1 ”Is the project aim to solely validate and/or verify a system?”

This node was created from mainly interviews with internal and external stakeholders. The external stakeholders explained that in cases where they have considered using optimisation, this question is commonly raised as a proposed system, i.e where the aim is to validate and/or verify, tend to not be well suited for optimisation. The internal stakeholders stated that a project with an aim such as this seems to provide the engineer with limited freedom, as many parameters may not be changed. These points led to ”*Is the project aim to solely validate and/or verify a system*” being the first question of the decision tree.

4.4.2 ”System design and input variables cannot be changed”

This node is not a question that the user should ask themselves, but rather a statement and a reminder to the user that in the majority of cases where this node is reached, the design of the system and the input variables cannot be changed. This statement is not always true, but in a vast majority of cases, which is why it is stated in decisive terms, to act as a strong reminder to the user. This statement was formulated with information gathered from the interviews with both internal and external stakeholders. Many of the interviewees stated that in cases like these, the engineer is quite limited in freedom as to what parameters may be changed. It was also stated by stakeholders that in cases like these, optimisation can lose some of its potential effectiveness.

4.4.3 "Are there decisions in the system that can be supported by optimisation?"

This node was mainly formulated through conversations with employees of AFRY Gothenburg. It was established that in some cases where the project aim is to validate and/or verify a system, the engineer may change certain variables, which leads to optimisation perhaps being more beneficial. In cases where this is not possible, there may still be decisions that can be supported by optimisation which can be found with some creativity from the engineer.

4.4.4 "Is it reasonable to believe that the real-life system will have optimisation?"

This question was formulated from external interviews. During interviews with the stakeholders possessing experience and knowledge in optimisation, it was stated by several of them how important it is to not design an optimised system that is smarter than its real-world counterpart. The question "*Is it reasonable to believe that the real-life system will have optimisation*", or slight variations of it, was raised by stakeholders as an important question to ask in situations like these, as an optimised system with a real-life counterpart without optimisation will have very different outcomes and not function as intended.

4.4.5 "Is it viable that optimisation can be implemented in decision-making?"

This question was formulated by talking to internal stakeholders. A previous project was brought up during a conversation about prior projects, with a rail with two mounted machines which allowed them to move sideways on the rail. The project aim was to verify whether the rail and the machines were sufficient for the expected workload. The manual problem solving process during this project led to the optimal solution being that the machines had to switch places with each other, which was impossible seeing as they were mounted on the same rail. This was then formulated as the question "*Is it viable that optimisation can be implemented in decision-making?*".

4.4.6 "System design and input variables can be changed"

This node was created from both interviews and conversations with employees of AFRY Gothenburg, and much like the first node on the left side, acts as a statement and a reminder for the user. During both interviews and when studying past projects, it was established that when the project aim is not solely to validate and/or verify a proposed system, the engineer can experience more freedom in regards to changes that can be made to the system.

4.4.7 ”Is there a foundation of known data, such as run-times or number of machines?”

This question was formulated from both external stakeholder interviews and conversing with employees of AFRY Gothenburg. The external stakeholders, who all have experience with optimisation, stated that in cases when the user has a significant amount of accurate and verifiable data, the optimisation has potential for more accurate results. This was mirrored in the secondary data, where several prior projects that provided access to large amounts of historical data had showed the internal stakeholders potential for optimisation.

4.4.8 ”Are variables dependant on each other?”

The question in this node was formulated from internal and external stakeholder interviews. During internal and external stakeholder interviews, it was stated that in projects where variables were dependant on each other, testing potential solutions can take a lot of time. In large systems, testing potential solutions can take an unreasonable amount of time, at which point the external stakeholders pointed out the usefulness of optimisation tools in such situations. Optimisation tools could streamline the testing process which could save time and money.

4.4.9 ”Is the amount of necessary experiments too large to perform manually?”

This question was formulated mainly from external stakeholder interviews. These stakeholders stated that systems with independent variables are generally less complex compared to that of a system with variables dependent on each other, and that the number of experiments increase if variables are dependent on each other. They stated that in cases with a large amount of necessary trials and complex systems can benefit from using optimisation, which could also lessen the workload of the user.

4.4.10 ”Is the effort of implementing optimisation of input variables larger than the return?”

This question was formulated from internal and external stakeholder interviews. Some of the external stakeholders raised the point that in some cases, it is possible to implement optimisation but it may not be worth it. The time spent implementing optimisation could be smaller than the expected return, in such cases it would likely not be worth it to implement optimisation. During internal discussions, i.e secondary data, the question was formulated to its current form, after contemplating what effort is necessary for the implementation in comparison to expected outcomes.

5
Discussion

The following chapter presents and discusses the methods used, the findings for the research questions, future work, limitations, and quality of research of this thesis.

5.1 Literature study

As mentioned in Section 3.1, the literature study included both recent and old articles. This is due to the fact that the foundation of DES has seen very few and minor fundamental changes since its inception in the middle of the 20th century, meaning the less recent articles explaining concepts around DES are still very much relevant to this day. One interesting aspect of using older articles as well as recent literature is to see how the understanding of DES has changed over time, as well as how the use of the technology has varied throughout the decades. We found that in earlier literature, the concept of DES was described in a simpler way, where as in the more recent articles there is a large focus on widening the scope of DES, which makes it more complex in comparison.

When an interesting article was found during the literature study, we found it useful to note the listed keywords used in that specific article, which could then be used to find similar papers or broaden the search of the literature study. References from particularly interesting papers were also studied, which provided the opportunity to delve deeper into any significant topics. A limitation of a literature study is that the keywords used might not be optimal. This risk is mitigated by investigating the references and keywords in relevant papers. This is a technique that was practised during this thesis, this provided us with a large set of data from the literature study that helped in answering the research questions.

5.2 Qualitative study

The qualitative study consisted of interviews with various stakeholders. The study was divided into internal stakeholder interviews and external stakeholder interviews where the goal and questions of the two differed. The main goal of the internal stakeholder interviews was to get an understanding of the current situation in AFRY Gothenburg's simulation team and provide insight in the interests of using optimisation with DES. On the other hand, the external stakeholder interviews was conducted to get insight of how DES and optimisation is used by distributors of software, academia, and other companies. An important distinction between internal and external stakeholders was that all of the external stakeholders have prior experience with optimisation while the internal stakeholders does not.

The external stakeholders were found from recommendations from our supervisor at AFRY as well as our supervisor at our university. We also contacted stakeholders that we made connections with through a prior course in DES at Chalmers University of Technology. The external stakeholders had varying degrees of experience and with their diverse backgrounds, we believe that they complement each others knowledge well. The internal stakeholders also have a varying degree of experience with working in DES, although the background of the internal interviewees are not as diverse in comparison with the external stakeholders.

When formulating the questionnaires for both the internal and external stakeholder interviews, we reviewed what was needed in order to accurately answer the stated research questions and what information we deemed necessary for a satisfactory completion of this thesis. It was important to us that the questions asked during the interviews were unbiased and that the queries were asked in a way that left room for interpretation by the interviewees while it still provided us with the data needed. While leaving room for too much interpretation might lead to the interviewees to answer entirely different questions, we aimed to enhance the fact that they came from different backgrounds with varied competences, which we believe was achieved by letting the interviewees relate the questions to their own background. All of the interviews were recorded in order to help in the transcription and analysis part of the qualitative study. It was therefore important to make sure that consent was given by the interviewee for a recording to be saved of the interview. These recordings and transcriptions are not included in this thesis for the sake of keeping the interviewees anonymous.

5.3 Answering the research questions

The following two subsections discusses how the two research questions of this thesis were addressed.

5.3.1 What optimisation methods are used in DES and what are the potential impacts of combining them with DES?

As of today, several DES software are capable of utilising optimisation methods. As previously mentioned, the three software investigated in this thesis; Plant Simulation, AutoMod, and FlexSim all contain built-in optimisation resources [55][56][57]. All of the mentioned software apply evolutionary algorithms in their optimisation calculations. Literature showed that EA is by far the most favoured algorithm for optimisation within DES [59]. The mentioned software also allows the user to bridge external, customisable, optimisation frameworks. Studies show that evolutionary algorithms are the norm in these types of scenarios as well. Evolutionary algorithms, as explained earlier in this thesis, does not ensure a global optimal solution to a stated problem but it is specialised in steering away from any local optima that other algorithms may tend to converge at [16]. This means that evolutionary algorithms converge toward the global optima which oftentimes is sufficient in DES. Simulated annealing is often mentioned in the same studies that score evolutionary algorithms as the leading algorithm in DES, and tends to be seen as a worthy contestant to EA, but it is very seldom implemented outside of research and academia. There does not seem to be any clear reasoning behind this, as the literature is consistent in showing that SA should, in theory, yield results that are near identical with those of EA. We think that this is a case of EA being tried and tested, and therefore there are no, or little, incentive to make alterations.

One of the more pressing questions regarding this matter is whether optimising a DES model actually yields a better result compared to manually improving the model. While it is difficult to quantify potential improvements leaning one way or the other, there are both advantages and disadvantages with optimisation in DES. One major benefit of using optimisation in DES is that it could ease the workload on the simulation engineer in cases where a large amount of experiments needs to be performed. In these cases, the user inputs the data and parameters that is to be varied which allows the algorithm to calculate without further supervision of the engineer. The engineer can therefore tend to other value-adding tasks while the algorithm performs the calculations, after which the engineer evaluates the results generated by the algorithm. Another benefit of using built-in optimisation tools in DES software is that it can lower the threshold for a user to make meaningful alterations which have an impact on the systems results. The qualitative study showed that the engineers heavily rely on experience when finding areas of improvement in their DES modelling. A less experienced user could utilise built-in optimisation tools to mitigate the advantage a more experienced user would have. It is important to keep in mind that although these tools may lower the threshold of improving a DES model, implementing them in a model has a varying level of difficulty depending on the DES software used. While the optimisation process could produce results that are superior to the ones produced by an engineer, this is not necessarily always the case.

It is possible that the engineer may figure out close to optimal solution without the need of optimisation methods and in these cases the time and money spent on optimisation calculations are not as justifiable. This relates to the fact that, as previously stated, optimising a DES model is always more expensive than not using optimisation [16][17]. The usage of optimisation tools requires extra work to be done, more in some software than others and most when building a customisable optimisation algorithm, only considering the extra time spent integrating optimisation in the DES model adds cost. The calculation resources required also result in increased cost that needs to be considered. However, the increased expenses can be justified. For example, if an existing real-world system is to be altered, simulating and optimising can identify solutions to problems that may otherwise have been overlooked. The optimisation algorithm could potentially increase the systems overall efficiency further, which although this comes at an increased cost, it is highly likely to be less expensive compared to altering the system and later realising that further changes are needed, prompting a second rebuilding phase. One should consider the risk of building an optimised model that outperforms, and is more intelligent than its corresponding real-world counterpart. If this is the case, the real-world system would not accurately represent the actions, choices, and results generated by the optimised model. The model could be rendered useless and resources spent on creating it would be in vain. This, of course, is also a risk to consider when one does not use optimisation methods in DES. A simulation model, whether optimisation is used or not, should never be smarter than its real-world counterpart.

5.3.2 What are the criteria for a DES case to qualify for optimisation?

As has previously been mentioned, a large portion of the simulation projects undertaken at AFRY Gothenburg are validation- and/or verification projects. Due to the nature of these projects, they are not always suitable for optimisation. This is because there, a client aims to validate or verify whether their planned design or alterations of a system will work as intended. In these kinds of projects, there is often no room for optimisation in conjunction with DES models, due to the system design and input variables oftentimes not being modifiable.

To help engineers answer the question of when optimisation is suitable to use, we created a decision tree that can be seen in Figure 5.1. The decision tree was mainly based on results from the qualitative study, but strengthened by literature and secondary data. Projects may look similar on the surface, but oftentimes there are differentiating factors that make them unique. It is therefore important for the user to relate and criticise each step of the decision tree to their own project query, boundary conditions, and preliminary study. A vital note to make is that this decision tree is to be used as a helping tool and should not be considered to be applicable in all scenarios, this is a consideration that we leave to the potential user. We encourage users to personalise the decision tree to fit their intents and purposes and as for academia to further develop and improve it.

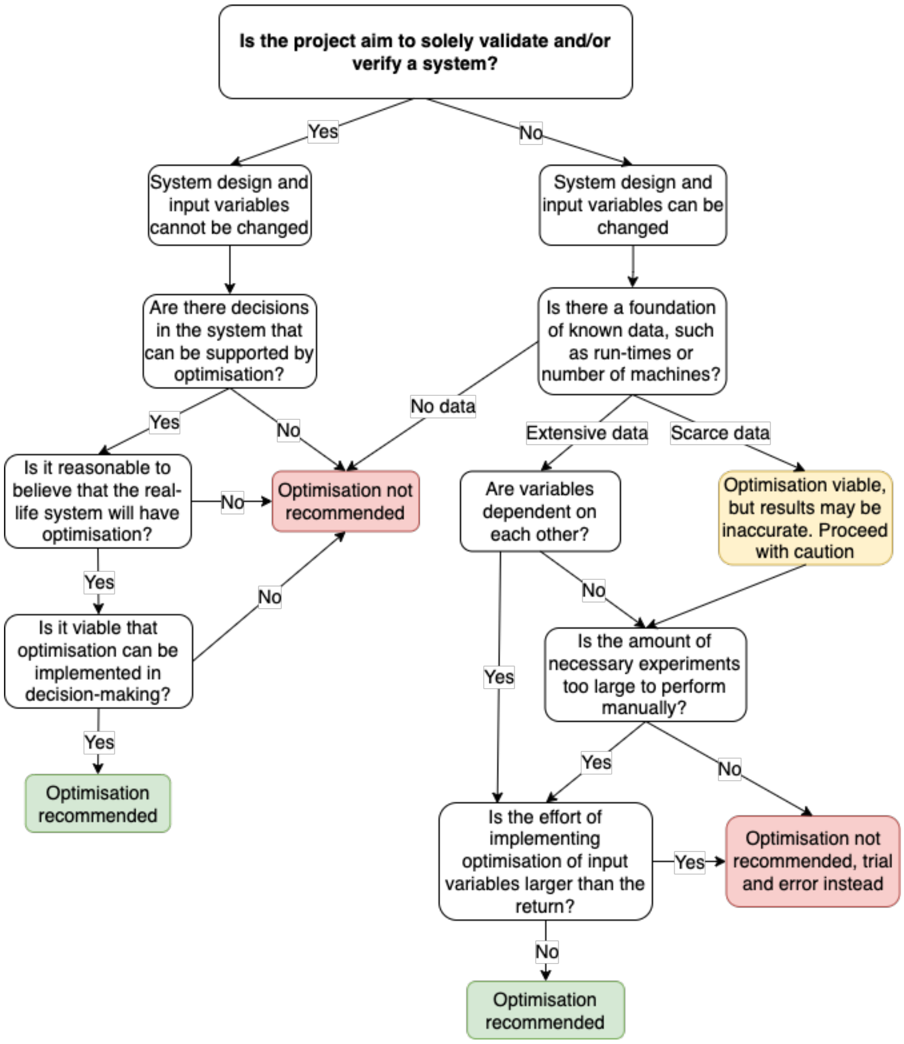


Figure 5.1: Decision tree to better understand when optimisation may be appropriate in discrete-event simulations

The following paragraphs explain each node and decision further in detail.

Is the project aim to solely validate and/or verify a system? - As an opening question, the user needs to look into the query and goal of the project and determine whether the aim is to validate and/or verify a system or not. As previously mentioned, a project where the aim is to validate and/or verify a system is often, according to us, not suited for optimisation unless certain criteria are fulfilled. In these projects it is not generally in the clients interests to further develop the real-life system in terms of layout, parameters and input variables. The user answers this question and follows the corresponding path, the upcoming four paragraphs explain the left side of the decision tree while the remaining five paragraphs describe the right side of the tree.

System design and input variables cannot be changed - This node is a statement that lets the user know that the conditions of system design and input variables moving forward. Since, in this case, the aim of the project is to validate and/or verify a system, there is often already an existing proposal that is to be evaluated. Therefore, changing the design and/or input variables could drastically alter the foundation of the simulated system, making the model redundant. As this node is a statement and not a question, there is only one possible outcome when leaving this node.

Are there decisions in the system that can be supported by optimisation? - In a scenario where the user is able to change the system design and input variables, those, among with many other parameters, could be optimised. As those are unable to be changed in this scenario, the user needs to consider if there are other parameters that could be optimised within the system, such as number of operators or sizes of buffers. If the user finds that there are parameters that can be changed, the path leads to further nodes to determine whether optimisation is still viable or not. However, if there are no parameters that can be changed, we do not recommend applying optimisation tools in this case, since it is likely to yield no meaningful results.

Is it reasonable to believe that the real-life system will have optimisation? - It is important for the user to consider if the real-life system will be capable of making the same decisions as the simulated optimisation model. There is a risk that a simulated system is smarter than its corresponding real-life system, and if this is true, optimisation is not recommended. This happens when the simulated system is capable of making decisions based on data that the real-life system is not able of doing. An example of this is operators, in a simulated system the operator has access to all variables, such as queues and buffer sizes, and takes decisions based on this. In the real-life system it is unreasonable to expect the operator to have access to all available information and make the same decisions as the simulated operator. If it is still reasonable to believe that the real-life system will be able to make the same decisions as the simulated system, proceed to the last question of this branch.

Is it viable that optimisation can be implemented in decision-making? - For optimisation to be of use, it is important that the user considers whether or not optimisation algorithm(s) can be implemented in the decision-making in conjunction with the simulation model. It is important to note whether or not there are decisions that a simulated system can make, that are not possible in real life. If there are decisions that are made in the real-life system that can be supported by optimisation, we recommend proceeding with optimisation.

System design and input variables can be changed - This node is a statement that lets the user know that the conditions of system design and input variables can be changed moving forward. Since parameters in the project can be changed, the possibilities of adapting optimisation tools are potentially increased. As this node is a statement and not a question, there is only one possible outcome when leaving this node.

Is there a foundation of known data, such as run-times or number of machines? - Having a foundation of data is vital when it comes to optimisation, as more accurate and extensive data can lead to a better result. Greenfield projects typically lack much of the valuable data that is important for optimisation. Brownfield projects, however, can often generate the necessary data based on experience from that site. If there is no available data, optimisation is not recommended. If there is scarce data, optimisation is still viable, but the results are at risk of being inaccurate and it is therefore suggested to proceed with caution. If there is extensive data available, there are good chances for the optimisation to be successful.

Are variables dependent on each other? - A system with the same amount of independent variables is generally less complex and thus the number of required experiments tends to be lower, compared to that of a system with those variables dependant of each other. In the latter case, the number of experiments that need to be evaluated drastically increase, since a change in one variable has implications on all, or some, variables. In the instance of dependant variables, we recommend that optimisation is implemented. If the variables are independent of each other, manual experiments might still be viable.

Is the amount of necessary experiments too large to perform manually? - When variables are independent of each other the number of experiments, and the experiments themselves, are generally less complex to evaluate in the DES software using trial and error, therefore excluding optimisation. However, if the amount of necessary experiment is too high, optimisation could be a viable tool. This, as previously mentioned, allows the user to spend their time on other value-adding activities instead of manually performing a large number of experiments.

Is the effort of implementing optimisation of input variables larger than the return? - As stated earlier, implementation of optimisation in DES is always more expensive and time consuming than not doing it, it is therefore important for the user to contemplate whether or not the effort of implementation would be larger than the potential return. If the effort is larger than the expected return, we do not recommend optimisation. Otherwise, we would recommend implementing optimisation.

5.4 Relation to previous work

As covered in the theory, simulation optimisation has been a recurring subject at the Winter Simulation Conference for more than two decades. Several articles and research papers has been published during these conferences for both the theoretical aspects of simulation optimisation and the practical implementation in a real-world scenarios. Some examples of this being the use of simulation optimisation in manufacturing[40][42][43], healthcare industry[47], logistical systems[39][44], and a more theoretical focused setting[49].

We do, however, believe that there is still a considerable gap between the proof of concepts shown during the Winter Simulation Conferences and the implementation in full scale industrial settings, specifically within the three software researched further in detail in this thesis. While it has been proven to work, and provided industry with successful examples and solutions of where and when it is applicable to implement optimisation in discrete-event simulation, we found through the qualitative study and the first literature study that it is yet to be fully applied and standardised in the work processes for simulation engineers. We believe that one reason for this could be the lack of a decision support tool to use during the implementation of optimisation in simulation, which we have created in order to start bridging the current research gaps introduced in Section 1.4

We did also find other interesting articles of relevance in previous work. One example of this, which is further covered in Section 5.5, is the hybridisation of two or more optimisation algorithms that has been mentioned previously[59]. There are many articles [16][18][22] showing high performance using various optimisation methods, mainly EA, despite this it appears that utilising optimisation is not as prevalent in industrial use as we had expected. Literature also showed that there does not seem to be any clear reason as to why evolutionary algorithms are more prevalent than others, such as simulated annealing algorithms [59]. This statement was supported by interviews with external stakeholders during this thesis, where the general consensus was that EA is "tried and tested" and there is no clear reason to change.

5.5 Future work

An interesting aspect to study further in future work is the opportunity of using artificial intelligence (AI) to improve optimisation processes within DES. AI models could be employed to develop existing optimisation models that continuously learn from both historical data and real-time feedback from applied systems. These models could potentially dynamically adjust their strategies based on changes in the systems, leading to more adaptive and robust optimisation solutions. A traditional optimisation algorithm relies heavily on historical data and static datasets, whereas an AI-driven optimisation tool could have the possibility to learn continuously from one or several systems and evolve over time. In a case of having a manufacturing system with constantly fluctuating demand and/or constraints, a continuously learning AI model could optimise production schedules in real-time to minimise costs and maximise efficiency.

Future studies within the choice of optimisation algorithm should be conducted as it is clear that there is a tendency to favourise the usage of evolutionary algorithms while studies show that other optimisation algorithms such as simulated annealing perform equally or even better than EA in certain cases. As we have previously mentioned, a hybridisation of optimisation methods may be a favourable path to consider when investigating types of optimisation. Oftentimes, no single optimisation algorithm is superior over all others. A combination, or hybridisation, of two or more algorithms may allow researchers and simulation engineers to develop algorithms that utilise the strengths from each of the stand-alone algorithms and mitigate their potential weaknesses. A hybridisation of optimisation algorithms, much like the previously mentioned customisable algorithms, require the simulation engineer to integrate multiple algorithms and components into a cohesive framework that works for the intended purpose. A task like this can take a long time to develop, debug, and fine-tune their code, whereas the built-in optimisation tools available in several DES software would require significantly less time spent on implementation and integration. This would be a question of priority which answer varies depending on the project; ”*Do we want the optimal result needing more resources or a satisfactory result with less effort?*”.

We believe that if an organisation aims to fully utilise hybrid systems together with DES models, an optimal solution would be to do it in a custom environment. As mentioned, the implementation and integration of hybrid optimisation algorithms with DES models adds complexity to a project. Simulation engineers have to establish strong and consistent communication between the optimisation algorithm and the simulation environment, which might be problematic due to simulation software limitations, where the user may not be allowed to extract and input all necessary parameters.

5.6 Quality of research

The literature study conducted in this thesis provided a multitude of academic papers, articles, and journals in the field of DES as well as optimisation. However, the authors did have difficulties in finding appropriate documentation regarding the adaption of using the two in combination. When searching for articles with keywords such as ”simulation optimisation”, ”DES optimisation”, or ”optimisation of simulation systems”, many results covered the act of optimising the computing efficiency while running a DES model. While this is a form of optimisation, it is the incorrect interpretation of the keywords for this use-case. There were articles that were found using the keywords mentioned above that accorded with the interests of this study but a majority of these papers are limited to academic research and theoretical cases while few showed actual implementation in real-world scenarios.

The sample size of the qualitative study was small, in particular the external stakeholder interviews. We believe that the qualitative study, despite its size, supplied us with plenty of relevant data in order to aid us in answering the research questions. There are negative aspects of having a small sample size, in many cases it is difficult to have interviewees with diverse backgrounds and viewpoints when handling fewer interviewees. However, there are also positives with smaller sample sizes as it allows the researchers to analyse the data from each interview further in depth. As previously mentioned, we do have some prior experience working with DES via our academic backgrounds. This could potentially cause bias and skew the results of the qualitative study, this was prevented by ensuring that the questions were asked in a neutral manner that did not lead the interviewee into answering the questions in a certain way. It should be noted that the course taken by us during our studies did not cover the topic of optimisation. The prior knowledge within DES proved to be helpful in understanding certain concepts during this thesis and paved the way for further comprehension.

To ensure a high quality of the research we decided to contact both the internal and external stakeholders from the qualitative study for feedback and critique on the thesis. Specifically, we asked whether or not they agreed with the decision tree, suggestions, and results that was created and received valuable points and considerations which were taken into account before finalising the thesis. To improve the quality of the work, we would recommend having a larger number of interviewees in the qualitative study in order to get a wider perspective. To further strengthen the results of this thesis it would be interesting to test implementations of optimisation in DES software, this was however not the goal of this thesis from neither the academic nor corporate stakeholders.

As previously described, all interviews were recorded and transcribed to allow us to extract the data gathered [53][54]. We did not face any notable issues during the transcription process, which could otherwise have compromised the quality of the data collected. We consider the triangulation used throughout this thesis to have increased its quality, since it allowed us to collect data and information from many different sources and methods. All information, both from the literature study and the qualitative study, was organised in documents throughout this thesis. All findings from the literature study was collected in a document which included, for each relevant reference, its title, access date, author(s), DOI, and a short description of its contents. As for the information gathered throughout the qualitative study, documents including all questions and their corresponding transcribed answers from all interviewees were used, separated between internal and external stakeholders.

6 Conclusions

The objective of this thesis was to identify areas where the combination of optimisation and simulation can be used effectively in order to support business decisions within production and intralogistic systems. By conducting a case study methodology and exploring the possibilities of utilising optimisation algorithms within DES software it was found that there are several software that provide built-in optimisation tools. This thesis focused on AutoMod, FlexSim and Plant Simulation, of which all three have built-in optimisation tools. It was also found that there are external optimisation tools that can be bridged to these software. The literature study and the qualitative study showed that evolutionary algorithms is the most common algorithmic method used in DES and it is ranked in the top in a majority of the studies done in the subject. Evolutionary algorithms is used because of its efficiency and ability to escape local optima, it does however not guarantee that it can find the global optima. The results are often satisfactory for the purpose of DES, but it should be noted that it does not necessarily supply the overall best solution for the real-world system.

The qualitative study provided the authors with the important insight that it is crucial to have the correct use-case in order for optimisation and DES to work together efficiently. A large portion of projects within DES is restricted to validation and verification of a real world system, where the room for modifications is minor. Therefore it can be important for the project to allow for changes to layout, parameters, buffer sizes, operators etc., in the real world system to benefit from optimisation. Results also show that different DES software have slightly varying ease of use for the simulation engineer. According to the qualitative study, AutoMod requires the user to keep optimisation in mind from the very start of the model building, while Plant Simulation required nothing extra from the engineer and could in most cases be applied to already existing models with little or no modification at all.

For optimisation to function efficiently it requires a lot of data, and for it to function accurately the data also needs to be verified and correct. Brownfield projects can often provide more accurate data that can be collected from the real-world system while Greenfield projects tend to have estimated data instead due to the real-world system still being in a planning phase. This could affect the results of the optimisation in the cases where the calculations are done on incorrect and flawed data, but can still be used as a guideline for the work ahead.

The authors provided AFRY with a decision tree that could potentially help simulation engineers to determine and identify whether or not their current project could benefit from the use of optimisation and DES. The decision tree should be used as a guideline for the engineer in their decision making process but also leave room for the engineer to make their own choices based on the recommendations provided by the decision tree.

A gap was found between research and industry where the studies proved that in general, optimisation with DES provides a better result but it is still not used to the extent it could in industry. It is also given that optimisation does come with an increase of cost and requires extra time for calculations and runs, this time does not necessarily need supervision by an engineer and thus they could spend time on other tasks during this. It would be interesting to see how the industry could continue to implement and utilise optimisation within DES and benefit from it.

Bibliography

[1] S. Robinson, “Discrete-event simulation: From the pioneers to the present, what next?” *Journal of The Operational Research Society*, vol. 56, Jun. 2005. DOI: `10.1057/palgrave.jors.2601864`.

[2] A. Goti, *Discrete Event Simulation*. Rijeka, Croatia: InTech, 2010, ISBN: 978-953-307-115-2.

[3] D. L. Olson, “Software process simulation,” pp. 143–153, 2003. DOI: `https://doi.org/10.1016/B0-12-227240-4/00163-5`.

[4] A. Skoogh, B. Johansson, and J. Stahre, “Automated input data management: Evaluation of a concept for reduced time consumption in discrete event simulation,” *SIMULATION*, vol. 88, no. 11, pp. 1279–1293, 2012. DOI: `10.1177/0037549712443404`.

[5] F. Malek-Abdel, C. Wolf, F. Johnson, and T. Spencer III, “Or practice: Survey results and reflections of practising informs members,” *J Oper Res Soc*, vol. 50, no. 10, pp. 994–1003, 1999. DOI: `10.1057/palgrave.jors.2600805`.

[6] S. Andradóttir and S. H. Kim, “Fully sequential procedures for comparing constrained systems via simulation,” *Naval Research Logistics (NRL)*, pp. 403–421, 2010. DOI: `10.1002/nav.20408`.

[7] D. L. Olson, “Optimization models,” H. Bidgoli, Ed., pp. 403–411, 2003. DOI: `https://doi.org/10.1016/B0-12-227240-4/00128-3`.

[8] N. Palm, Personal Communication, Jan. 30, 2024.

[9] AFRY, *About us*, Available at `https://afry.com/en/about-us`, Accessed: 2024-01-25.

[10] S. Robinson, *Simulation: The Practice of Model Development and Use*, 2nd ed. New York: Bloomsbury Publishing, 2014.

[11] T. J. Schriber and D. T. Brunner, *Inside Discrete-Event Simulation software: How it works and why it matters* (Winter Simulation Conference). New York NY: ACM Press, 1997, vol. 3204. DOI: `https://doi.org/10.1145/268437.268441`.

[12] S. Amaran, N. Sahinidis, and B. Sharda, “Simulation optimization: A review of algorithms and applications,” pp. 301–333, 2014. DOI: `10.1007/s10288-014-0275-2`.

[13] J. Banks, *Discrete-Event System Simulation*, 4th ed. Pearson, 2005.

[14] A. M. Law and D. W. Kelton, *Simulation Modeling & Analysis*, 2nd ed. Singapore: McGraw-Hill, 1991.

[15] IBM, *What is optimization modeling?* Available at `https://www.ibm.com/topics/optimization-model`, Accessed: 2024-01-29.

[16] G. Venter, "Review of optimization techniques," 2010. DOI: 10.1002/9780470686652.eae495.

[17] A. Griewank and A. Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation* (Other Titles in Applied Mathematics), 2nd ed. Society for Industrial and Applied Mathematics, 2008, ISBN: 9780898716597.

[18] R. G. Parker and R. L. Rardin, *Discrete Optimization*. London: Academic Press, Inc, 1988.

[19] P. A. Vikhar, *Evolutionary algorithms: A critical review and its future prospects*. 2016, pp. 261–265. DOI: 10.1109/ICGTSPICC.2016.7955308.

[20] J. C. Spall, *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, Inc., 2003. DOI: 10.1002/0471722138.

[21] T. Bäck, *Evolutionary Algorithms in Theory and Practice*. New York: Oxford University Press, 1996.

[22] A. E. Eiben and J. E. Smith, *Introduction to evolutionary computing*. Springer, 2015.

[23] S. Sieniutycz and J. Jeżowski, *Energy Optimization in Process Systems*. Amsterdam: Elsevier, 2009, pp. 1–43, ISBN: 978-0-08-045141-1. DOI: https://doi.org/10.1016/B978-0-08-045141-1.00001-9.

[24] M. A. Cervinski, A. Bietenbeck, A. Katayev, T. P. Loh, H. H. van Rossum, and T. Badrick, "Advances in clinical chemistry patient-based real-time quality control (pbrtqc)," *Advances in Clinical Chemistry*, vol. 117, pp. 223–261, 2023. DOI: https://doi.org/10.1016/bs.acc.2023.08.003.

[25] Y. Eren, İ. B. Küçükdemiral, and İ. Üstoğlu, *Optimization in Renewable Energy Systems*. Boston: Butterworth-Heinemann, 2017, pp. 27–74, ISBN: 978-0-08-101041-9. DOI: https://doi.org/10.1016/B978-0-08-101041-9.00002-8.

[26] Y. C. Hi, M. H. Lin, J. F. Tsai, and C. S. Yu, "A review of deterministic optimization methods in engineering and management," *Mathematical Problems in Engineering*, vol. 2012, no. 756023, 2012.

[27] S. G. Nash, *Nonlinear programming*, Available at https://www.informs.org/ORMS-Today/OR-MS-Today-Software-Surveys/Nonlinear-Programming-Software-Survey/Nonlinear-Programming, Accessed: 2024-02-15.

[28] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming* (Other Titles in Applied Mathematics), 5th ed. Springer Charm, 2021, ISBN: 978-0-387-74502-2.

[29] S. G. Nash and A. Sofer, "Linear and nonlinear programming," 1996.

[30] K. Eisemann, *Linear Programming*. International Business Machines Corporation, 1955.

[31] G. B. Dantzig, "Linear programming," *Operations Research*, pp. 42–47, 2002. DOI: https://doi.org/10.1287/opre.50.1.42.17798.

[32] M. Schulze, "Linear programming for optimization," Sep. 2000.

[33] R. Kunwar and H. Sapkota, "An introduction to linear programming problems with some real-life applications," *European Journal of Mathematics and Statistics*, vol. 3, pp. 21–27, Apr. 2022. DOI: 10.24018/ejmath.2022.3.2.108.

[34] G. B. Dantzig, "Linear programming," *Operations research*, vol. 50, no. 1, pp. 42–47, 2002.
[35] J. S. Arora, *Introduction to Optimum Design*, 5th ed. Academic Press, 2023, ISBN: 9780128183205.
[36] D. S. Chen, R. G. Baston, and Y. Dang, *Applied Integer Programming: Modeling and Solution*. John Wiley & Sons, 2011, ISBN: 9781118210024.
[37] D.-Y. Chan, C.-Y. Ku, and M.-C. Li, "A method to improve integer linear programming problem with branch-and-bound procedure," *Applied Mathematics and Computation*, vol. 179, no. 2, pp. 484–493, 2006. DOI: `https://doi.org/10.1016/j.amc.2005.11.171`.
[38] *Winter simulation conference archive*, Available at `https://informs-sim.org/` (2024/06/10).
[39] D. Schmaranzer, R. Braune, and K. Doerner, "Simulation-based headway optimization for a subway network: A performance comparsion of population-based algorithms," *Winter Simulation Conference*, 2018. DOI: `https://doi.org/10.1109/WSC.2018.8632362`.
[40] J. Bernedixen, M. Andersson, S. Bandaru, and T. Lezama, "Aircraft assembly ramp-up planning using a hybrid simulation-optimization approach," *Winter Simulation Conference*, 2020. DOI: `https://doi.org/10.1109/WSC48552.2020.9383887`.
[41] G. Hura, "Petri net as a modeling tool," *Microelectronics Reliability*, vol. 22, no. 3, pp. 433–439, 1982, ISSN: 0026-2714. DOI: `https://doi.org/10.1016/0026-2714(82)90022-1`.
[42] Y. Wang and A. N. Chen, "Production logistics simulation and optimization of industrial enterprise based on flexsim," *Winter Simulation Conference*, 2016. DOI: `https://doi.org/10.2507/IJSIMM15(4)CO18`.
[43] L. Bobberman and T. Vink, "Applications of simulation optimization to minimize downtime," *Winter Simulation Conference*, 2021.
[44] S. G. Henderson and N. Jian, "An introduction to simulation optimization," *Winter Simulation Conference*, 2015. DOI: `https://doi.org/10.1109/WSC.2015.7408295`.
[45] K. Chen, "Poisson process," in Feb. 2015, pp. 141–151, ISBN: 9781848217478. DOI: `https://doi.org/10.1002/9781119006190.ch8`.
[46] B. Verweij, S. Ahmed, A. Kleywegt, G. Nemhauser, and A. Shapiro, "The sample average appromixation method applied to stochastic routing problems: A computational study," *Computational Optimization and Applications*, 2003. DOI: `https://doi.org/0.1023/A:1021814225969`.
[47] T. M. Lal, T. Roh, and T. Huschka, "Simulation based optimization: Applicatons in healthcare," *Winter Simulation Conference*, 2015. DOI: `https://doi.org10.1109/WSC.2015.7408251`.
[48] D. Linz, B. Ahmadi, R. Good, E. Musabandesu, and F. Loge, "Multi-threaded simulation optimization platform for reducing energy use in large-scale water distribution networks with high dimensions," *Winter Simulation Conference*, 2020. DOI: `https://doi.org/10.1109/WSC48552.2020.9383920`.
[49] M. Zhang, A. Matta, A. Alfieri, and G. Pedrielli, "Simulation-based benders custs: A new cutting approach to approximately solve simulation-optimization

problems," *Winter Simulation Conference*, 2018. DOI: https://doi.org/10.1109/WSC.2018.8632326.

[50] R. Rahmaniani, T. G. Crainic, M. Gendreau, and W. Rei, "The benders decomposition algorithm: A literature review," *European Journal of Operational Research*, vol. 259, pp. 801–817, Jun. 2017. DOI: 10.1016/j.ejor.2016.12.005.

[51] R. Yin, *Case Study Research: Design and Methods* (Applied Social Research Methods), 4th ed. SAGE Publications, 2009, ISBN: 9781412960991.

[52] M. Pidd, "Computer simulation for operational research in 1984," *Developments in Operational Research*, pp. 19–30, 1984. DOI: 10.1016/b978-0-08-031829-5.50007-5.

[53] Microsoft, *Transcribe your recordings*, Available at https://support.microsoft.com/en-us/office/transcribe-your-recordings-7fc2efec-245e-45f0-b053-2a97531ecf57, Accessed: 2024-05-16.

[54] Microsoft, *View live transcription in microsoft teams meetings*, Available at https://support.microsoft.com/en-us/office/view-live-transcription-in-microsoft-teams-meetings-dc1a8f23-2e20-4684-885e-2152e06a4a8b, Accessed: 2024-05-16.

[55] Z. Vidrova, Z. Bohus, and P. Adamko, "Simulation as a tool for process optimization in a manufacturing company," *SHS Web of Conference*, vol. 92, 2021, ISSN: 2261-2424. DOI: https://doi.org/10.1051/shsconf/20219208023.

[56] Applied Materials, *Automod user's guide*, Santa Clara, California, USA, 2020, Accessed: 2024-04-13.

[57] Autodesk, *FlexSim User Manual*, 2017, Accessed: 2024-04-14.

[58] P. Raska and Z. Ulrych, "Testing optimization methods on discrete event simulation models and testing functions," *Procedia Engineering*, vol. 69, pp. 768–777, 2014, 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013, ISSN: 1877-7058. DOI: https://doi.org/10.1016/j.proeng.2014.03.053. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877705814002999.

[59] C. L. Renzi, F. Cavazzuti, and A. O. M. Andrisano, "A review on artificial intelligence applications to the optimal design of dedicated and reconfigurable manufacturing systems," *The International Journal of Advanced Manufacturing Technology*, vol. 72, pp. 403–418, 2014, ISSN: 1433-3015. DOI: https://doi.org/10.1007/s00170-014-5674-1.

[60] H. Smedberg, C. A. Barrera-Diaz, A. Nourmohammadi, S. Bandaru, and A. H. C. Ng, "Knowledge-driven multi-objective optimization for reconfigurable manufacturing systems," *Mathematical and Computational Applications*, vol. 27, no. 6, 2022, ISSN: 2297-8747. DOI: 10.3390/mca27060106. [Online]. Available: https://www.mdpi.com/2297-8747/27/6/106.

A

Appendix 1

Question	**Goal**
What is your title at AFRY?	To get the title of the employee
How long have you been in your current role, and do you have any previous experience in DES?	To get an understanding of their professional experience
What software(s) do you primarily use for DES?	Find the commonly used DES softwares at AFRY
What proportion of your projects are improvement/optimisation projects, e.g. improving the output of a factory?	Find the proportion of projects that optimisation tools may be suitable for
How do you arrive at potential solutions/improvements in your work?	To get an understanding of how the employee currently handles these types of tasks
How much time do you spend testing solutions/improvements in projects?	To get an understanding of how their time is spent in such a project
Do you feel a need to be able to use optimisation tools in your work? If so, in what type of projects or queries?	To find whether the employee can see a use case for optimisation tools
What would it take for you/your team to start using optimisation in conjunction with your simulation models?	Find what the employee need to potentially start using optimisation tools
Are you aware of any available optimisation tools in the software(s) you use regularly?	Find if the employee knows of any optimisation tools
Have you tested any of the optimisation tools available in PlantSim, FlexSim, and/or AutoMod etc.?	To identify whether the employee has used any available tools
Is there anything else you would like to add to the subject that we have spoken about?	Asked to give the employee a chance to bring forth their own opinions

Table A.1: Interview questions and their respective goals for internal stakeholders

B
Appendix 2

Question	**Goal**
What is your job title?	To get the title of the interviewee
How long have you been working with DES and optimisation tools?	To get an understanding of their professional experience within the field
What is your preferred DES software?	Identify which software the interviewee primarily works with
Have you used optimisation tools outside of DES?	Identify areas where the interviewees have used optimisation in other applications
Do you, or have you personally use(d) the optimisation tools available in PlantSim/AutoMod/FlexSim?	To determine the interviewee's exposure to the available optimisation tools
In what cases do you consider it appropriate to use optimisation tools?	Identify types of projects and/or queries where the use of optimisation tools may be useful
Do you know the extent to which the optimisation tools in your preferred DES software are used?	Understanding whether the available tools are being used
PlantSim/AutoMod/FlexSim all have optimisation tools and the same method is employed in all of them, evolutionary algorithms. Why do you think this is?	Understand why major DES softwares use evolutionary algorithms
If you compare someone with extensive knowledge in DES with the built-in optimisation tools, what differences and similarities can you expect from both results?	Understand the differences and similarities between how a problem is solved by both a user and the optimisation tools
If a simulation engineer spends 100 hours on a simulation model and manual improvement, approximately how much time would the optimisation tools need to improve the model in this case?	Identify how much extra time (if any) is needed to utilise the optimisation tools
Does the simulation engineer need to provide anything additional when building the model to use the optimisation tool? Do you need to consider optimisation when building the simulation model?	To find if the user would need to change their modelling and workflow to adapt to the optimisation tools

Table continued from previous page

Question	**Goal**
Is it optimised iteratively, or is it satisfied with the first result it reaches?	Further understanding of the software in question
Do you have good control over how the optimisation tools can change the model?	Further understanding of the software in question
Have you heard of any other optimisation methods used in DES? External programs, bridging to Python etc?	To get a broader understanding of the tools available on the market
Is there anything else you would like to add in the subject?	Asked to give the interviewee a chance to bring forth their own opinions

Table B.1: Interview questions and their respective goals for external stakeholders

DEPARTMENT OF INDUSTRIAL AND MATERIALS SCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY