Sequencing and Launching of Trucks with Varying Lengths on a Paced Moving Mixed-Model Assembly Line

Graduation Thesis at Scania Production Zwolle

Jedidja Visser, BSc
Industrial Engineering & Management

SCANIA

UNIVERSITY OF TWENTE.

This thesis is written as part of the graduation assignment of the master programme of Industrial Engineering & Management at the University of Twente.

University of Twente
Industrial Engineering and Management
Postbus 217
7500 AE Enschede
Tel. 053 4899111

Scania Production Zwolle
Production Engineering
Russenweg 5
8041 AL Zwolle
Tel. 038 4977611

Sequencing and Launching of Trucks with Varying Lengths on a Paced Moving Mixed-Model Assembly Line

Publication date: 21st of October, 2022
Number of pages: 65
Number of appendices: 3

Author:
Jedidja Visser, BSc
Master student Industrial Engineering & Management, University of Twente

Supervisors University of Twente:
Dr. ir. M.R.K. Mes
(1st supervisor)
Dr. ir. J.M.J. Schutten
(2nd supervisor)

Supervisor Scania Production Zwolle:
M. Damman, MSc
Groupleader production engineering

This thesis is intended for Scania Production Zwolle and the supervisors from the University of Twente involved in this research. In this public version, some parts are moved to a confidential appendix.

Preface

This report is the result of my graduation project at Scania Production Zwolle, which I executed for the final part of my Industrial Engineering & Management master programme. In this programme, I got many opportunities to develop myself, not only academically, but also professionally. For example, I had the opportunity to participate in the research honours programme, which made me interested in the world of academics. I also got the possibility to do a capita selecta and learn about research within the healthcare sector. Besides, I was able to join the organisation of a conference and offer teaching assistant support to courses. I feel that teachers always believed in me, and I am grateful that I was able to study at the University of Twente.

I am thankful that I could perform my graduation project at Scania Production Zwolle. I have always been interested in factories, and I really enjoyed walking along the assembly line and learning about the process of assembling a truck. Specifically, I want to thank Marien, who offered great support in the entire project. You helped me to always think critically, you gave very helpful feedback and supported me beyond this project. I also want to thank my team, the front office engineers, for all the motivation, interest and coffee I received.

Furthermore, I want to thank my supervisors from the University of Twente, Martijn en Marco. Martijn, you encouraged me to think out of the box in this project and stimulated me to keep improving my models and techniques. Marco, your feedback helped me to improve my writing, and thank you for your expertise in algorithms and assembly lines.

Finally, I want to thank my fellow students, roommates, friends, student association, boyfriend and family for supporting me the past six years. I have learned a lot in these years, about operations research and about myself, and I look forward to what the future will bring!

Jedidja Visser
Zwolle, October 2022

Management Summary

Problem Definition

This research is performed at Scania Production Zwolle (SPZ), which assembles trucks on a constant paced, moving assembly line, with a fixed takt time per work station. However, the truck lengths vary, thus the occupation time of a work station differs per truck. Therefore, we formulate the research question:

How can we maximize the output of a fixed speed assembly line with products with varying lengths?

In the current situation, a truck that is longer than the length of a work station causes output loss, because its following truck is launched later on the line, which cannot be compensated anymore. SPZ also uses light sensors that discretely measure and therefore overestimate the length of long trucks, which results in more output loss. The planning department currently includes the lengths of the trucks only implicitly by planning a maximum of one long truck in every three trucks, but it does not consider the time a truck occupies a work station because of its length.

Solution Design & Evaluation

We design solution approaches for all three analysed problems:

Measuring method: We formulate a discrete measuring location model, based on the P-median location model, which places p sensors such that the extra measured length of the trucks is minimized.

Launching strategy: In the current situation, a truck is only launched on the line at least one takt time after its preceding truck. We design a *triple takt strategy*, in which each station has one longer takt time for three combined trucks. If one of these trucks is a long truck, its two following trucks are allowed an earlier launch to compensate for the long truck, such that the total time for the three trucks does not exceed three takt times. We also design a *variable rate launching strategy*, where trucks are always allowed to be launched earlier on the assembly line, as soon as there is a sufficient distance to its predecessor.

Sequencing method: We formulate a genetic algorithm (GA), a metaheuristic based on evolution, to generate a sequence that optimizes the production rate of the assembly line. We design three approaches: a *discrete-event simulation (DES)* approach that focuses on evaluating the quality of the solutions, a *simplified evaluation (SE)* approach that deterministically evaluates the quality of the solutions to focus on the optimisation and a *simheuristics (SH)* approach that combines the first two approaches.

To evaluate our solutions, we develop a discrete-event simulation (DES) model, which simulates the assembly of a sequence of trucks and incorporates stochasticity. The key performance indicator (KPI) of this model is the average daily production rate of the assembly line.

Results

We execute a one-factor-at-a time analysis in our DES model to evaluate all designed alternative solutions:

Measuring method: Our DES model shows that if the light sensors are optimally placed according to our discrete measuring location model, the current production rate could be increased by 0.10% (statistically insignificant). This is the case from four added light sensors (9 light sensors in total). If the truck lengths are measured continuously, the production in the current situation can increase by 0.13%.

Launching Strategy: The variable rate launching strategy can increase the current production significantly, with 3.2%. The triple takt strategy can significantly increase the current production by 0.6%.

Sequencing method: The SE approach is not able to evaluate the quality of sequences accurately compared to the DES model. Therefore, we execute the DES approach, which performs best using an order crossover and inversion mutation, and the SH approach, which gives similar results to the DES approach with a faster run time. For all launching strategies, the differences between the current sequence and the GA generated sequences are insignificant. All these sequences result in a high production rate, and have in common that long trucks and trucks with high processing times are often sequenced close together.

Our sensitivity analysis shows that the average daily production rate could be increased most when boundaries are open, and the technical stoppage occurrence is low. The variable rate launching strategy responds less to these changes than the current and triple takt strategy.

Practical Contribution

The practical contribution of this research is that SPZ is able to increase their production rate by 3.2% if the variable rate launching strategy would be applied, and the lengths are measured continuously (if the lengths are measured discretely, the increase is 3.0%). If the triple takt strategy is applied, which is slightly easier to implement in a Lean-focused environment, the production rate can still be increased by 0.6%. This research also provides a discrete measuring location model, which SPZ can use to (re)position light sensors. Furthermore, we contribute a DES model of the assembly line, which is also applicable for testing other alternative solutions.

Scientific Contribution

We show that a sequence that results in a high production rate, often plans a long truck and a truck with high processing times (*complex truck*) close together, because long trucks can create extra space on the line such that the operators have more time to work on a complex truck in their work station. Furthermore, we show that it is complex to model a mixed-model assembly line with full-line stoppages in a deterministic, simplified simulation, since a simplified simulation cannot properly consider the impact of a stoppage on another simultaneous stoppage at another work station. Therefore, this study required a genetic algorithm with a discrete-event simulation to evaluate the quality of a sequence, which has a long run time to slowly optimise the sequences. For further research, we therefore suggest studying the improvement of the deterministic, simplified simulation and the development of simheuristics.

Contents

Chapter 1

Problem Introduction

This research took place at Scania Production Zwolle, as a graduation assignment of the master programme Industrial Engineering & Management at the University of Twente. This chapter introduces the problem that this research concerns. Section 1.1 discusses the company background and the motivation of this research. Section 1.2 describes the research plan, including the problem description, scope and research questions.

1.1 Company Background & Research Motivation

Section 1.1.1 introduces the company, Scania Production Zwolle. Section 1.1.2 gives the motivation of this research, and Section 1.1.3 provides the context of the analysed problem.

1.1.1 Scania Production Zwolle

Scania AB is one of the leading manufacturers in trucks, buses, engines and services. Scania was founded in Sweden in 1891 and is part of Volkswagen Group since 2014. Scania has about 50,000 employees in approximately 100 countries, where research and development is mainly based in Sweden and production facilities are located in Europe and Latin America. An important concept of Scania is their modular system, which allows customers to fully customize their trucks, which are then produced according to the Scania Production System (Scania CV AB, n.d.). This system is based on the Toyota Production System, but adapted to the Swedish culture (Oudhuis & Tengblad, 2020).

The biggest final assembly plant of Scania, Scania Production Zwolle (SPZ), is located in the Netherlands, which assembles around 210 trucks on a daily basis. SPZ produces trucks on two assembly lines, Castor and Pollux, which are both highly flexible to handle the modular system.

1.1.2 Research Motivation

In SPZ's strategy for 2022, goals are stated to reduce the stop times of the assembly lines and to decrease work related accidents (Scania Production Zwolle, 2022). SPZ would like to increase the daily production capacity to 240 trucks, while maintaining the flexibility of the assembly lines, which corresponds to mass customisation (Alford et al., 2000; Gilmore & Pine II, 1997). Because of this customisation, SPZ produces trucks with a varying length, where some lengths exceed the length of a work station on the assembly lines. Since the conveyor system of the assembly lines has a fixed, constant speed, long trucks occupy work stations longer. This is not in line with the first design rule of a lean manufacturing system, which states that every process should be balanced to the *takt time*: the available production time divided by daily demand (Black, 2007). If process times do not correspond with the takt time, this can lead to underproduction or overproduction. In SPZ, long trucks therefore lead to underproduction, and thus a waste of resources, which is not in line with lean manufacturing and results in output loss. The production engineering department of SPZ feels that there is still room for improvement in dealing with long trucks on their assembly lines.

The objective of this research is to minimize the output loss caused by long trucks on the assembly lines, in order to contribute to the increase of the daily production capacity, in the context of mass customization.

1.1.3 Problem Context

Figure 1.1 provides an overview of the two assembly lines at SPZ, with the Castor assembly line in more detail. Technically, both assembly lines can assemble almost all truck configurations, but since Castor has a shorter takt time, most heavy and many long trucks are scheduled on the Pollux line. The assembly lines are divided into two parts, with two work stations in between, as shown in Figure 1.1. Both lines start with a *stop & go* system (Part 1, until Station 27A on the Castor), where the frames are hanging in an overhead conveyor. In a stop & go system, the products are placed in a work station during one takt time, and then all synchronously moved to the next work station. In between the two parts, the frames are placed on a still-standing carrier at Station 28 and then synchronously moved to Station 29 where the engine is placed in the frame. After Station 29 onwards, Part 2 of the assembly line, the carriers start driving with a constant speed through the work stations until the end of the line. Section 3.1.2 further explains the concept of synchronous and moving assembly lines.

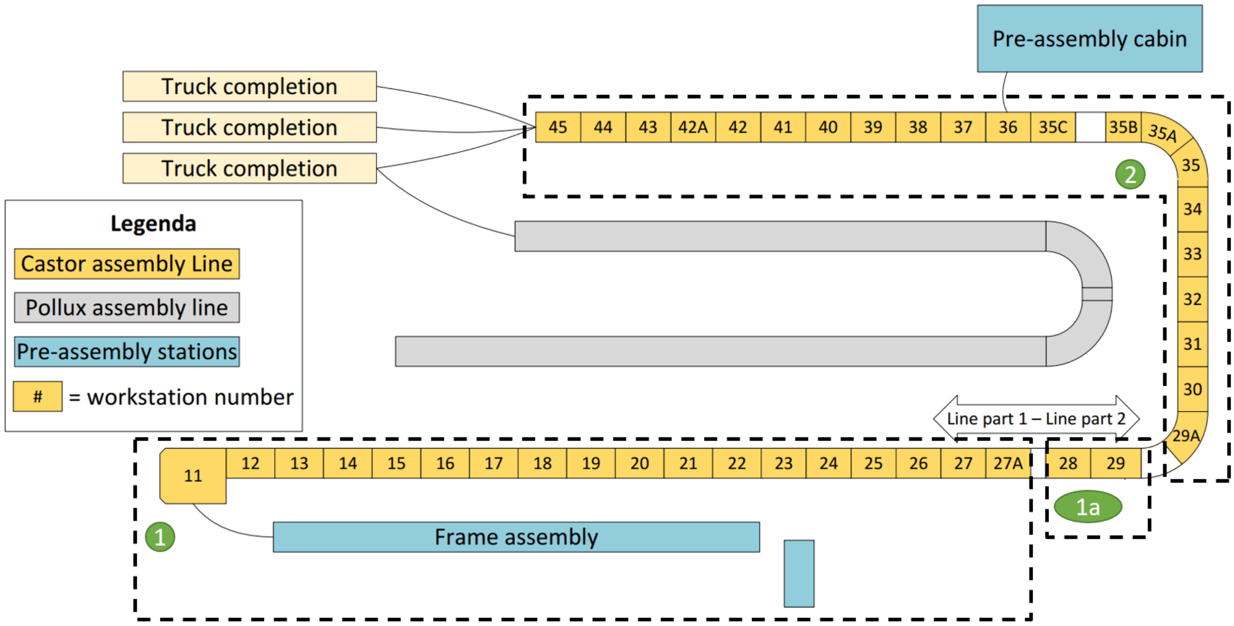


Figure 1.1: Overview of assembly lines SPZ (Juurlink, 2021)

We define a long truck as a truck whose length, including the *safety distance*, exceeds the length of a work station. This safety distance (see Figure 1.2) is based on the space needed to safely walk between trucks, and work in the front of the truck. Besides, sufficient space is needed for the cabin which is tilted for the assembly and the relatively sharp U-turn of the Pollux (see Figure 1.1). Since the carriers have no sensors, insufficient safety distance could lead to unsafe situations.

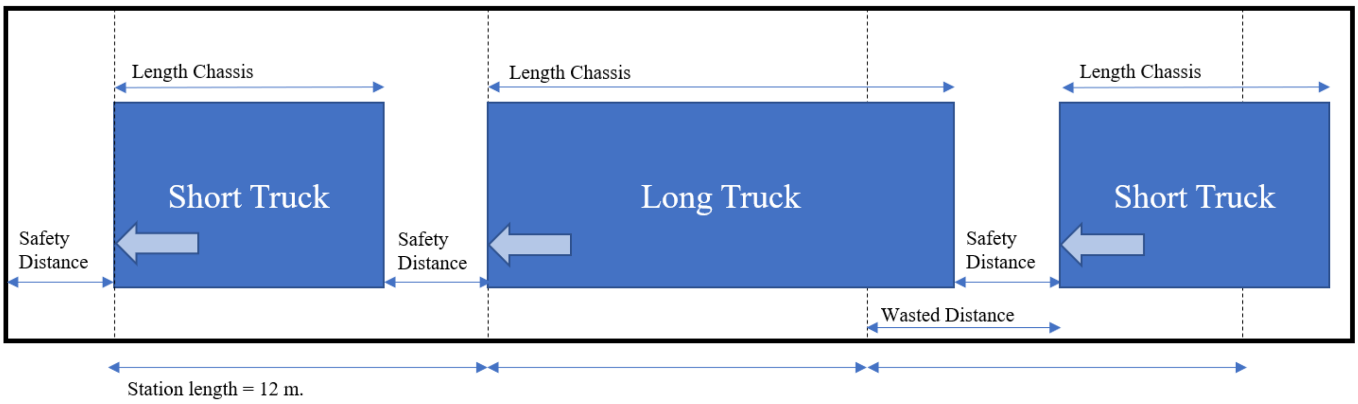


Figure 1.2: Trucks with varying lengths sequenced on assembly line

Takt times are an essential concept of SPZ's production system. Takt time is defined as the available product assembly time per work station that is needed to match the daily demand, see Equation (1.1).

$$\text{Takt Time (TT)} = \frac{\text{Available Production Time per Day}}{\text{Daily Demand}} \tag{1.1}$$

To match all work stations along the final assembly line with this takt time, all tasks are designed with processing times under the takt time. Therefore, there is no under- or overproduction if there are no disruptions (Black, 2007; Suh et al., 1998), since the daily demand is constant. The fixed speed of the carriers is based on the work station length (12 meter) and the takt time. After every takt time, a new carrier with a truck is launched on the line, which means that a new carrier can only start driving if the front of the preceding carrier is at least 12 meter on the line.

Since consecutive carriers need to keep a distance of at least the safety distance plus the length of the truck, the length of the truck is measured between the stop & go system and the constant speed carriers (Station 28 on Castor) with light sensors. The engineering department of SPZ decided to use this sensor method, because the actual length of a truck could vary from the length stated in the planning system, e.g., if a tow bar is assembled incorrectly. So, only taking the length from the planning system into account could lead to unsafe situations. Furthermore, the carriers can only handle a limited set of driving distances, so a discrete measurement method makes sense. Figure 1.3 provides a simplified illustration of Station 28 with 3 light sensors (there are 5 sensors in practice). The system labels the length of a truck according to the position of the first sensor that did not detect the truck. In the illustration of Figure 1.3, the distance until the succeeding carrier would be $length_{station} + (x_3 - x_1)$.

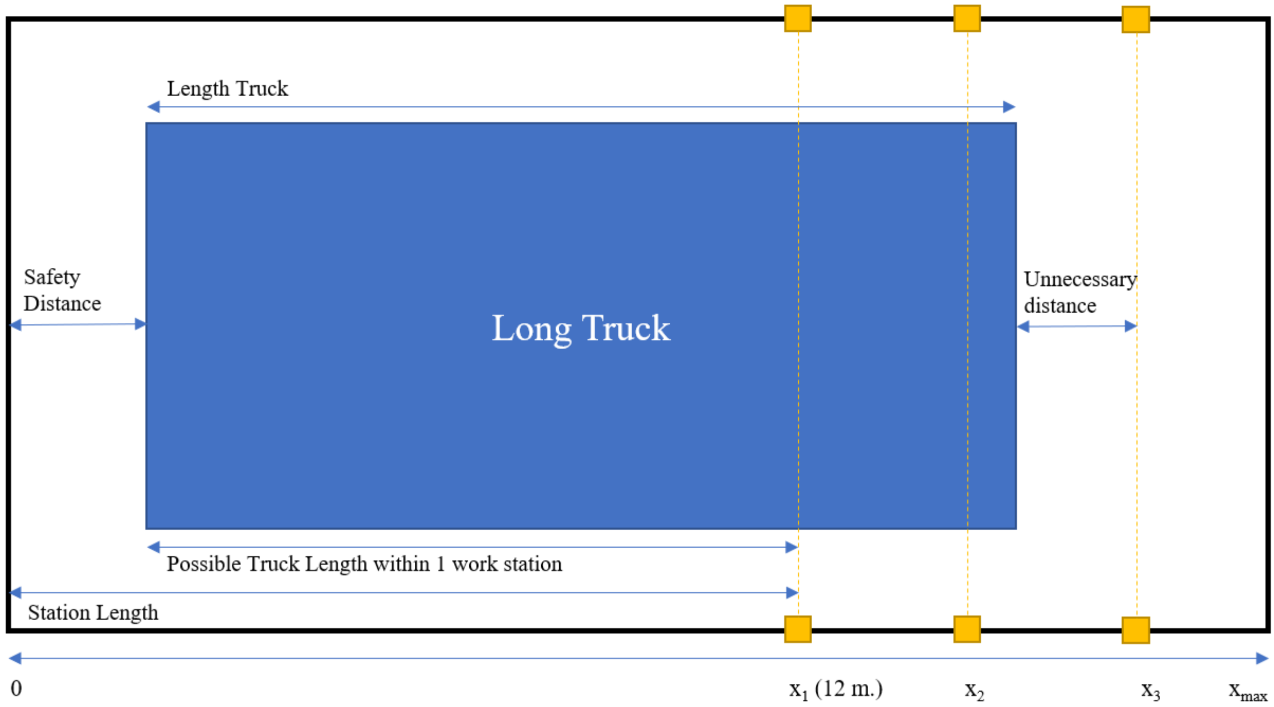


Figure 1.3: Simplified illustration of light sensors measurement

1.2 Research Plan

Section 1.2.1 discusses the impact of long trucks and trucks with variable lengths in general, and the root causes of this impact. Section 1.2.2 gives the scope and objective of this research, and Section 1.2.3 describes the research design. Section 1.2.4 provides the outline of this thesis corresponding to the research design.

1.2.1 Problem Description

The long trucks impact both parts of the assembly lines differently.

In the first part of the line (*stop & go system*), long trucks protrude in the previous work station, such that the succeeding truck's position is further back in the work station. If a truck protrudes too much, the fixed equipment at a work station cannot reach the truck anymore, which leads to stop time and thus an output decrease. Because of this, the production planning department attempts to prevent consecutive long trucks in the planning.

In the second part of the line, the carriers have a fixed speed. A long truck therefore occupies each work station longer compared to a truck that fits within one work station length, while a long truck does not necessarily need a higher processing time. Figure 1.2 shows this: a long truck drives through a work station taking longer than one takt time, resulting in a delayed start for the succeeding truck in that work station. The delayed start means that the resources at that work station have some idle time and that the assembly line cannot deliver one truck each takt time exactly, which is not desirable. In the current system based on the Toyota Production System philosophy, only one truck per takt time can be launched on the carrier line. This means that SPZ decided that shorter trucks are not allowed to compensate for the output loss of long trucks. Theoretically, this means that long trucks get a higher takt time, which Juurlink (2021) visualized (see Figure 1.4). This figure firstly provides a situation where tasks per work stations have different processing times, which are then balanced according to the Toyota Production System, explained in more detail in Section 3.1.2. However, long trucks on a continuous driven line result into fluctuating takt times, which disturbs the balance.

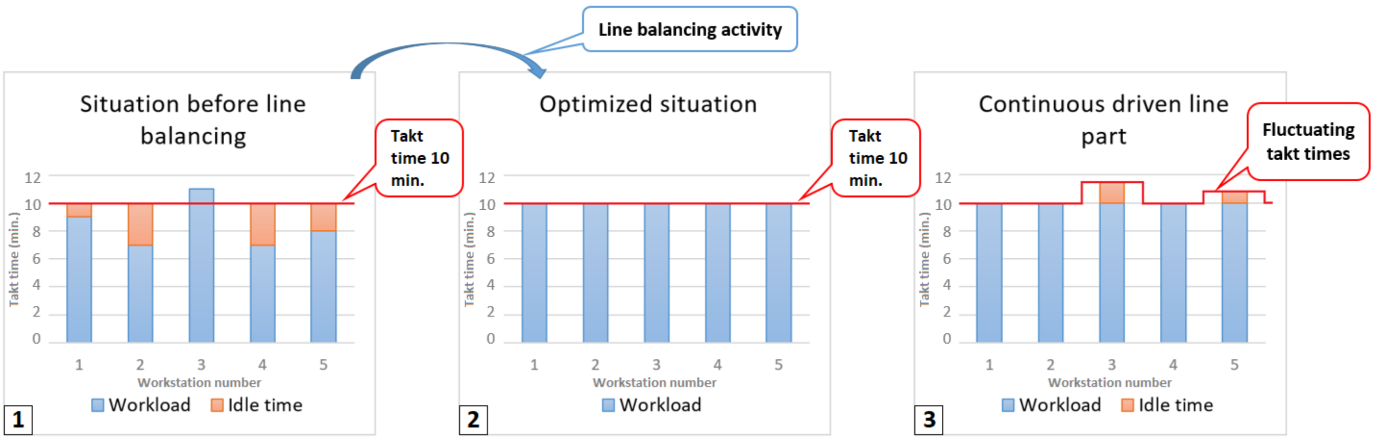


Figure 1.4: Long trucks result in higher takt times and idle time (Juurlink, 2021)

In addition to the delayed start of trucks because of the lengths of long trucks, the discrete measuring system of these lengths delays the start of trucks even more. As shown in Figure 1.3, the system always programs the distance to the next carrier given the worst-case scenario, because the sensors cannot detect with how much the truck exceeded the sensor (x_2 in the illustration), so it always takes the length of the next sensor into account. This results in a longer distance between consecutive carriers than necessary, which leads to wasted distance on the assembly line.

We schematically show all the discussed problems caused by these long trucks in the problem cluster in Figure 1.5. In this figure, the blue boxes show problems that are directly perceived by SPZ, the yellow boxes describe root causes that cannot be changed within this graduation project and thus are out of scope, and the green box represents a problem that is already solved in another graduation project, for which we refer to Juurlink (2021). The orange boxes represent problems that could be analysed in our research: the discrete measuring system of the long trucks; the fixed launching rate of one takt time; and the mixing rules of the production planning.

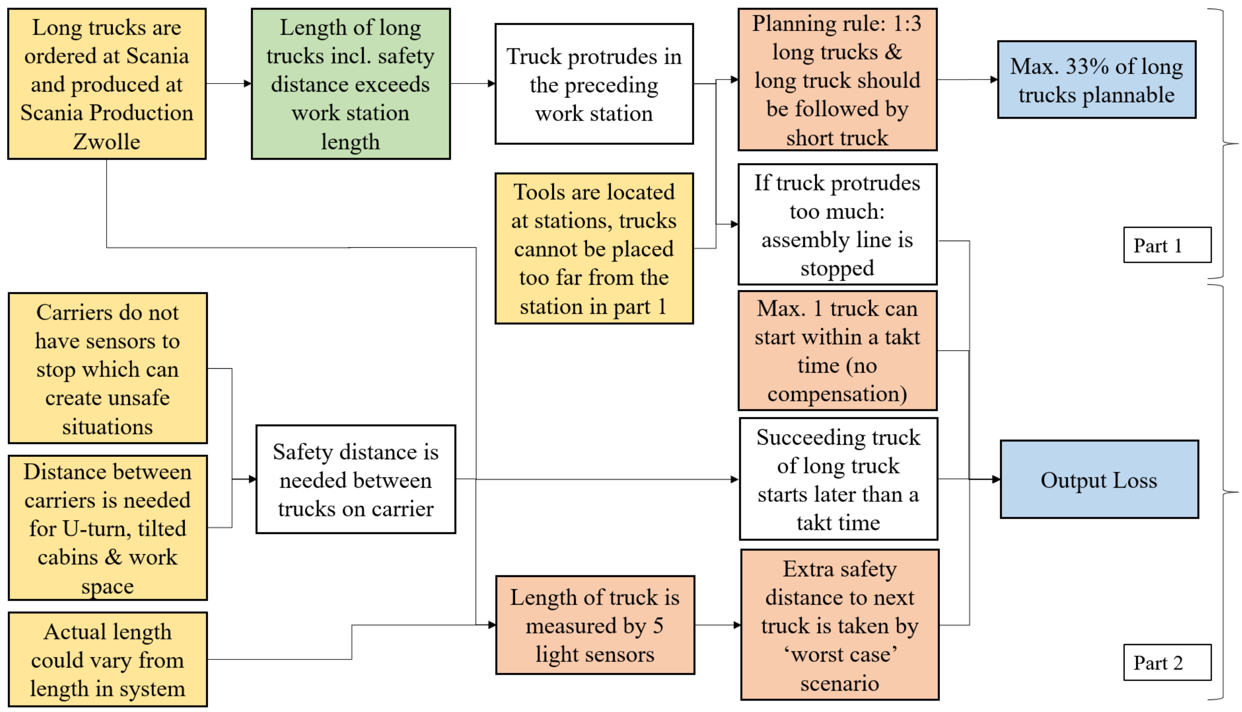


Figure 1.5: Problem cluster regarding long trucks

In general, the problems perceived by SPZ are not only caused by long trucks, but by the fact that all trucks have a variable length, while the assembly line has a fixed speed and a fixed takt time. The variable length gives trucks a variable occupation time of work stations, while the tasks are designed such that they all fit within a fixed takt time. Figure 1.6 shows the general representation of the problems in this research, using the same colour scheme as Figure 1.5. The yellow boxes show restrictions of this research that are out of scope, the blue box represents the problem as perceived by SPZ and the orange box describe problems that are tackled in this research.

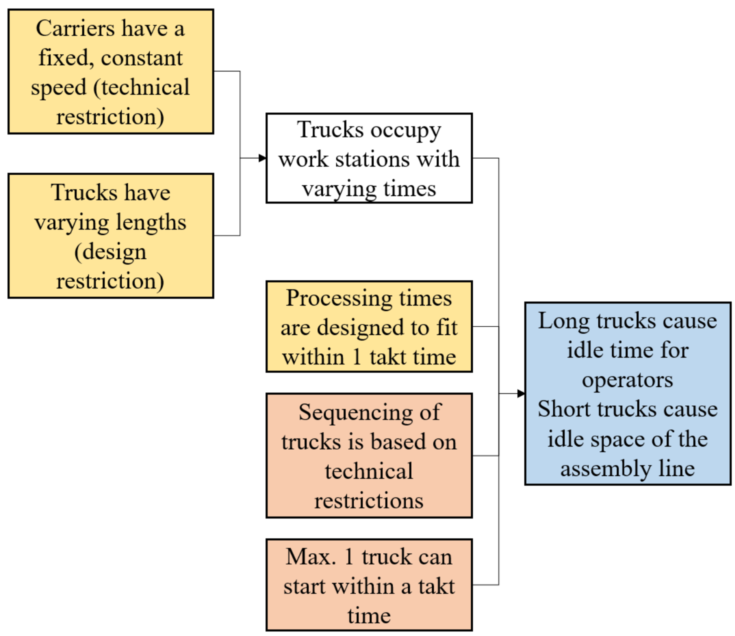


Figure 1.6: General problem cluster of trucks with varying lengths on a paced moving mixed-model assembly line

To conclude, we look into an assembly line with a fixed, constant speed, where trucks with varying length are assembled. The core problem that we identify is: *there is a discrepancy between the variable occupation time of work stations because of the variable length of the trucks on a constant moving assembly line, and the fixed takt time.* This problem is owned by the engineering and production planning departments of SPZ.

1.2.2 Research Objective & Scope

The objective of this research is to maximize the production output, while dealing with trucks with varying length on an assembly line with a constant speed. This objective contributes to the strategy of SPZ for 2022 to increase the number of assembled trucks per day and to decrease the stop times on the assembly lines (Scania Production Zwolle, 2022). From the problem analysis and problem clusters in Figures 1.5 and 1.6, we concluded that there are three solution approaches we look into to optimize the production output. We study the launching system of the trucks on the line, the sequencing of the trucks, and the discrete measuring system of long trucks.

Figure 1.5 provides the causes (in yellow) of the impact of long trucks that are out of scope regarding alternative solutions, such as the production planning decision by Scania Sweden, the safety considerations and the design of the trucks. In this project, we focus on the second part of the assembly lines with constant speed carriers, so we do not look into alternative solutions for the first part of the assembly lines. For the evaluation of alternative solutions, we take the first part of the assembly lines into account, since this does affect the performance of a possible solution. This research focuses on the assembly line Castor, since this line produces the highest volumes and thus can yield the most results. We focus on optimizing the existing lines, not on re-designing the core of the manufacturing processes. This means that the balancing of the lines, which focuses on assigning tasks to work stations, is out of scope, but the sequencing of the trucks is in the scope of this project.

1.2.3 Research Design

This section describes the research questions, which together form the design of this research, graphically shown in Figure 1.7.

To solve the core problem in this research, we formulate the main research question: *How can we maximize the output of a fixed speed assembly line with products with varying lengths?* To answer this question, we formulate the following sub-questions:

1. How does SPZ currently deal with trucks with varying lengths on their assembly line with fixed speed and what is the performance of the current situation?

To improve the output, we first have to understand the current situation at SPZ. We also determine performance indicators, such that we can compare alternative solutions with the current situation in this research.

2. Which solutions can be found in literature that create flexibility on an assembly line to deal with varying products?

If the current situation is known, we look into alternative solutions that follow from literature that relate to maximizing output on a mixed model assembly line.

3. What are alternative solutions that can be implemented at SPZ and how can we evaluate these solutions?

We design alternative solutions based on solutions found in literature such that they can be implemented in the specific case of SPZ. We also design an evaluation method to measure the performance of these solutions.

4. What is the performance of these solutions and what are the advantages and disadvantages?

For the solutions that we found in literature and that seem reasonable and possibly effective, we analyse these solutions and their impact on SPZ. To come to a good recommendation, we determine their performance, but also their advantages and disadvantages for SPZ.

5. Which recommendations can we give to SPZ to maximize their output given the fixed speed assembly lines?

If all possible solutions are analysed, we are able to give a recommendation to SPZ which solution, or combination of solutions, maximizes their production output.

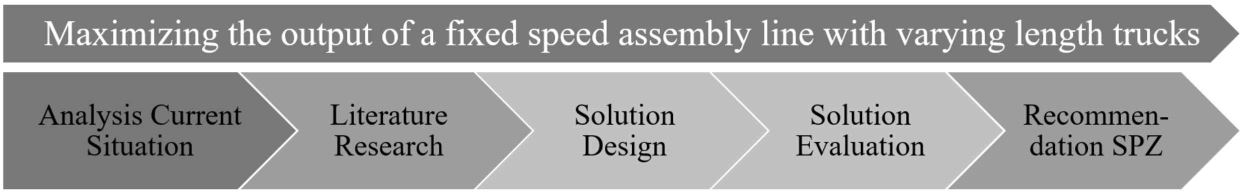


Figure 1.7: Research outline

1.2.4 Outline Thesis

This thesis follows the outline as shown in Figure 1.7. Chapter 2 focuses on the current situation of SPZ and its performance. Chapter 3 gives a theoretical framework for this research and discusses alternative solutions that are found in literature. These solutions are then analysed and discussed in Chapter 4. Chapter 5 designs an evaluation method for these solutions, and Chapter 6 presents the results. Chapter 7 concludes this research with a recommendation for SPZ and a conclusion and discussion on the methodology of this research.

Chapter 2

Current Situation

This chapter describes the current situation at SPZ regarding trucks with variable lengths on a fixed speed assembly line. Section 2.1 gives a detailed description of the assembly line Castor and Section 2.2 explains the production planning process. Section 2.3 presents a data analysis on the lengths of the trucks and the related problems.

2.1 Description of Assembly Line

Although Section 1.1.3 already briefly described the assembly lines at SPZ, this section further introduces the assembly lines. As explained in the scope of this research, we focus on the assembly line Castor, the high volume line.

The assembly line Castor consists of 41 work stations and multiple pre-assembly work stations. The first part of the line consists of a synchronous hanging conveyor system, which moves all chassis synchronously to the next station after each takt time. At these stations, the chassis can move vertically, such that tasks can be performed at all positions of the frame. Then, at station 28 (see Figure 1.1), the chassis is placed on a still-standing carrier and the length of the chassis is measured using the light sensors as described in Section 1.1.3. The takt time of this station is 80 seconds shorter than the general takt time of the entire assembly line, to create some independence between the first and second part of the line. This is also called the *harmonica mechanism*, which means that a stop at the first part of the line does not immediately force a stop on the second part of the line, and vice versa. After the shorter takt time, the carrier is moved to station 29, where the carrier is still not in motion yet. Here, the engine is installed in the frame. The carrier is then launched on the carrier line, starting from station 29A, if the following criteria are met: (i) the engine is installed successfully, (ii) the takt times of station 29 and 29A are finished, (iii) station 28 is finished (to ensure a good flow), and (iv) the rear of the preceding truck at station 29A is at least a safety distance away. From station 29A, the carrier drives in a constant motion, where operators walk along the carrier to execute their tasks. After finishing their tasks, the operators walk back to the start of their station. The speed of the carriers is set such that the front of a truck can pass a station in exactly one takt time.

If tasks are not finished at a station or if technological problems occur, the line is stopped. At the first part of the line, this means that all trucks do not move to the next station until the problem is solved. At the second part of the line, a stop results in halting all the carriers. The operators will still continue their tasks if the truck is already in reach of their equipment. Since the management focuses on avoiding stops as much as possible, the team leaders of the operators at the carrier line will often choose to finish a task at the next station if the equipment allows this, instead of stopping the line. If there is a failure, either in design, engineering or assembly, there is no stop, but a flexible operator will take over and join the truck until the failure is solved.

2.2 Production Planning

This section describes the production planning process, both on tactical and off-line and online operational level.

On the tactical level, Scania Sweden determines per *data period* which trucks are assembled in which plants, based on strategic planning decisions. Each data period consists of one week, in which an *order mix* should be assembled in that week in a plant. The number of trucks in this mix is determined by the number of days in a week multiplied by the daily production goal.

Then, on the off-line operational level, the production planning department at SPZ decides which trucks should be assembled at which assembly line. Scania uses car sequencing, which means that the sequence dependent *work overload*, which occurs if the task times exceed the available time in a work station, is minimized using implicit sequencing rules. Such rules, also called mixing rules, are formulated as $H_0 : N_0$, which means that out of N_0 subsequent trucks, only H_0 can have a specific trait (Boysen et al., 2009).

They use a genetic algorithm that determines the sequence of trucks per data period per line, that takes the mixing rules into account. An example of a mixing rule at SPZ is that out of 3 trucks, only 1 truck can be long. Besides these mixing rules, the algorithm also includes rules that require specific configurations (such as long trucks) to be well-spread over the sequence, and desired finished assembly dates. Each rule has a priority ranking, and the objective of the algorithm is to minimize the violations of the mixing rules, which are weighted with these priority rankings.

On the online operational level, a truck can be delayed because of missing parts or a design mistake. If these trucks are not on the line yet, these trucks are then pulled out the sequence and manually placed later in the sequence. If the trucks are on the assembly line already, the operators will still do their tasks as much as possible, but these trucks will be finished later, either by returning on the assembly line or by a repair team.

2.3 Data Analysis

Section 2.3.1 introduces the collection of the data which we use for an analysis and our evaluation model in Section 5.1. Section 2.3.2 analyses data on the truck lengths, and Section 2.3.3 analyses the output loss resulted by long trucks.

2.3.1 Data Collection

The available data at SPZ relevant for this research consists of the order information of the trucks, starting from the introduction of the new truck generation (NTG), the current product family. This family was introduced in May, 2018 and December, 2017 for Castor and Pollux, respectively. The data is collected until March 2022 inclusive. The calculation of the length of the trucks is based on a combination of product characteristics and design information, which is discussed with the responsible product engineers. Here, we assume that the product characteristics did not change during the assembly period of the NTG.

We also analyse the stoppages at the Castor assembly line, for which we use data that should not be influenced by management decisions to have a production goal of less than 100%, which happened a lot in the last years due to Covid-19 and material shortage. Therefore, we decided to only use data on June, 2021 for this analysis, because this is the only month in the last two years with production goals of 100% and no big stoppages.

2.3.2 Truck Length Analysis

To give context to our research, we analyse the distribution of truck lengths in the data set and look for possible trends. Figures 2.1 and 2.2 show the distribution of the trucks lengths per month over the NTG period of the assembled trucks at the Castor and Pollux line, respectively. The figures show that Castor line assembles a higher volume than Pollux, while Pollux assembles relatively more long trucks, as shown by Figure 2.3, which presents the distribution of lengths of the trucks over their entire NTG period per assembly line. The last periods, which were subject to Covid-19 and material shortages, show a lot of variability in production volumes and distribution. However, we can derive from the distribution of lengths of the trucks per month, that the amount of longer trucks seem steady over the periods. Therefore, we can use historic data for our experiments.

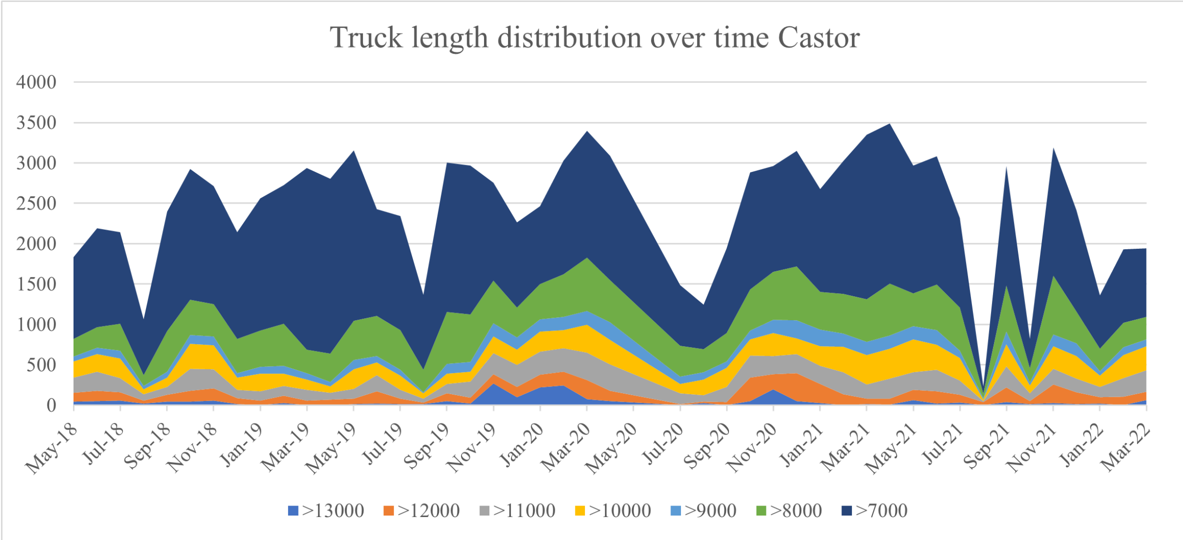


Figure 2.1: Length distribution over time at Castor

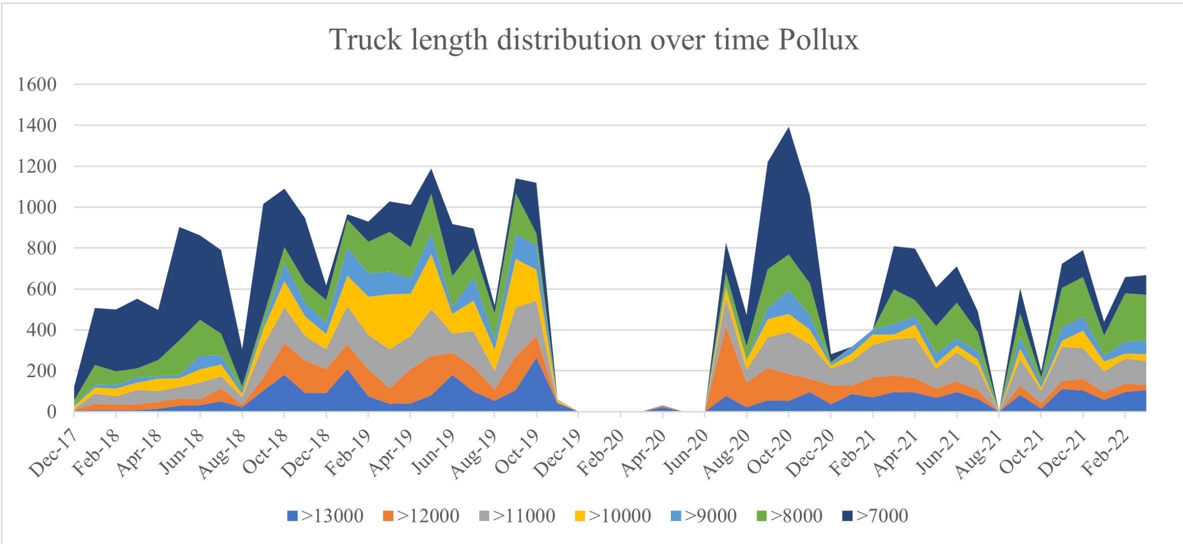


Figure 2.2: Length distribution over time at Pollux

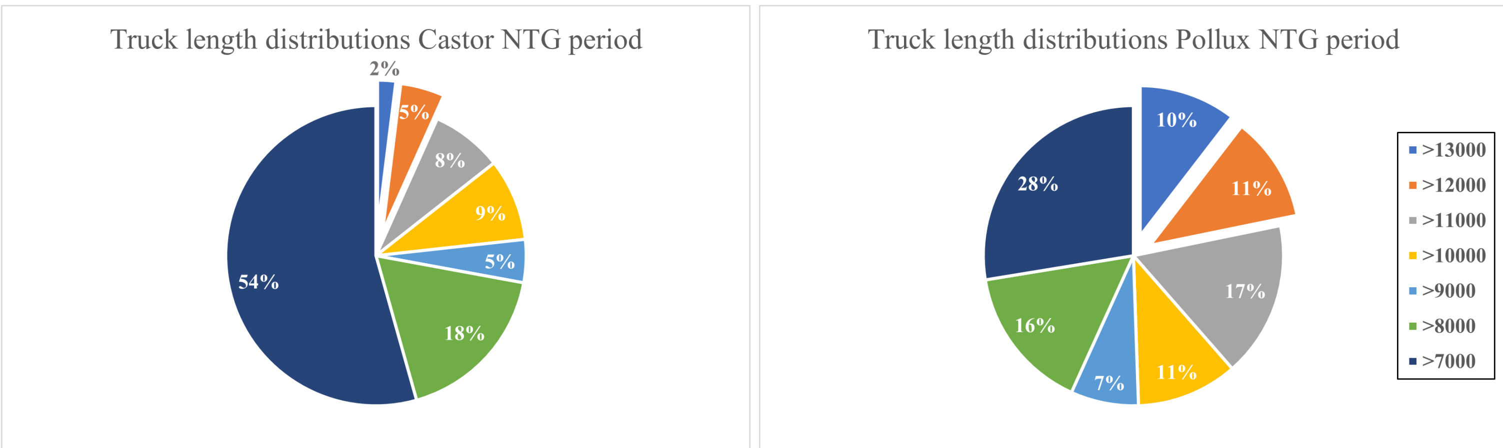


Figure 2.3: Length distributions Castor vs. Pollux over NTG period

2.3.3 Impact of long trucks

Long trucks result in output loss, both because of their length and because of the discrete measuring system, as shown in Figure 2.4. This section firstly looks into the total impact of the length of long trucks, and then into the output loss caused by the light sensors.

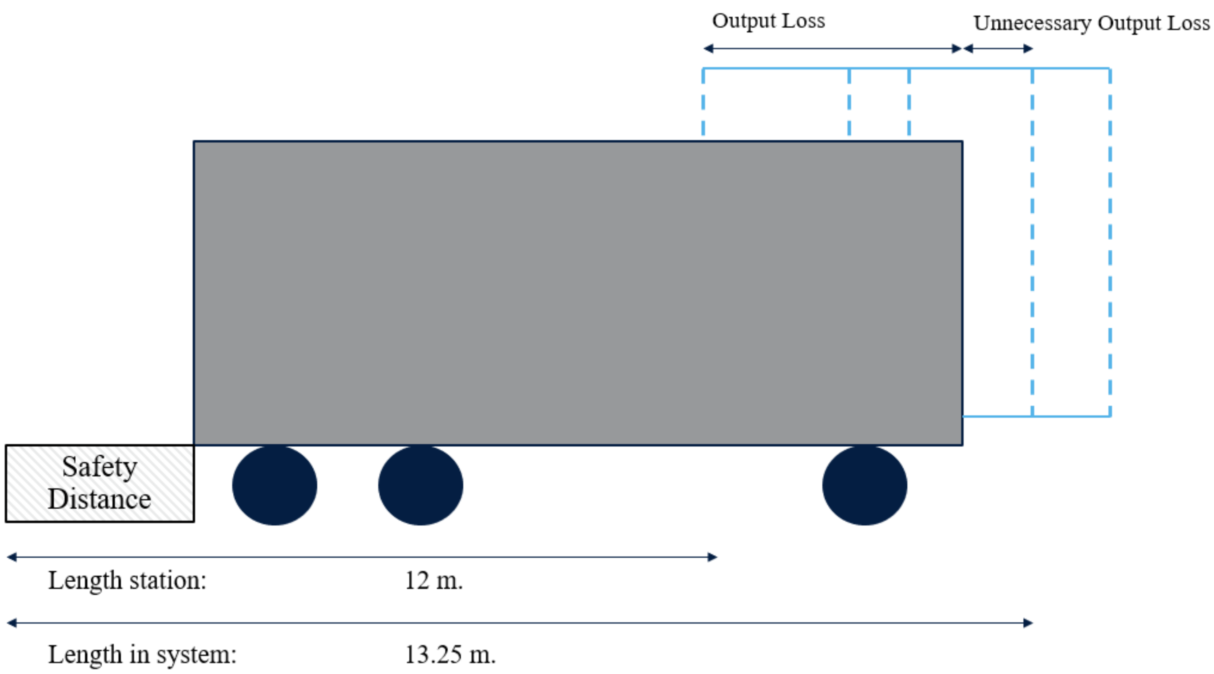


Figure 2.4: Output loss because of long trucks in current situation

For every standstill of the assembly line at SPZ, the cause of this standstill, its causing work station and its length are recorded. SPZ also automatically records the time a truck arrives later than the takt time at the end of the assembly line, if it follows a long truck. Although these trucks do not cause an actual stoppage of the assembly line, we use these records to calculate the percentage of time that the assembly line is 'stopped' because of long trucks. Table C.1 in Appendix C shows this percentage for June, 2021, including the stoppage percentages for both parts of the Castor assembly line. In the month June, the distribution of the length of the trucks is quite similar to the distribution of the overall period, based on the comparison between Figures 2.3 and 2.5. Although many engineers at SPZ have analysed the impact of long trucks on the assembly line, this data suggests that these long trucks do not create much stoppage time.

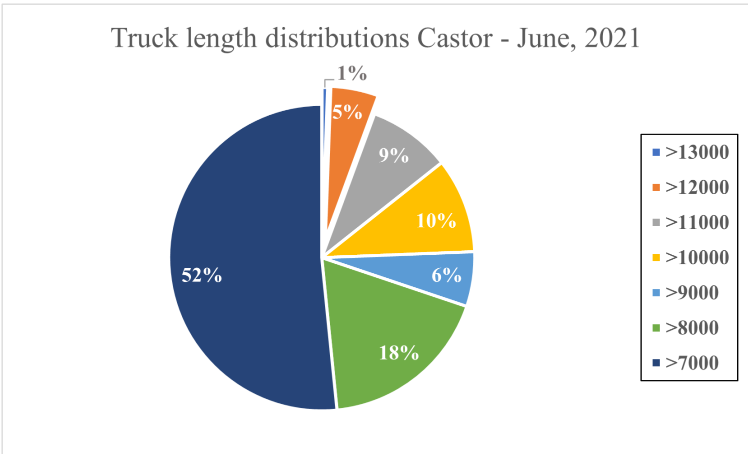


Figure 2.5: Length distributions of Castor in June, 2021

As Figure 2.4 shows, each long truck leads to unnecessary output loss because of the discrete measuring system and general output loss because of its length. Figure 2.6 shows a histogram that includes all lengths of long trucks assembled in the entire NTG period, using a bin size of 25 mm. The first five dotted lines represent the current positions of the light sensors, the last dotted line represents the maximum distance that the system records for trucks that are detected by all sensors. From this figure, we conclude that the position of light sensors does not always correspond with a high frequency of trucks in a specific length.

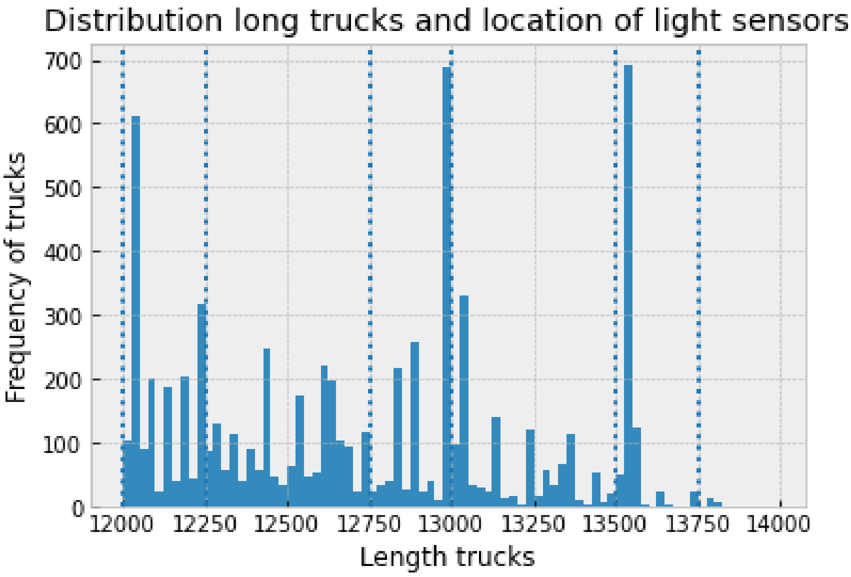


Figure 2.6: Histogram with the lengths of long trucks (Castor, NTG period) and positions of light sensors

Using the data of truck lengths in June, 2021, we can calculate the extra space that could not be used on the assembly line, both because of the length of the trucks and the measuring system. Then, we can calculate the missed number of trucks that could have been assembled on the extra space occupied by long trucks. Table C.2 in Appendix C shows the output loss of long trucks, expressed in meters, takt times, idle time (comparable to stoppage times) and costs. We show this for the month June, 2021, to compare this with the real-life data of this month in Table C.1, which shows that the idle time percentages because of long trucks are relatively similar. Although the stoppage percentage is quite small, from the costs we conclude that we can optimize the assembly line at SPZ.

2.4 Conclusion

To conclude, the current positions of the light sensors cause unnecessary output loss, where we discovered an improvement possibility. Besides the measuring system, the long trucks cause even more output loss, so we can optimise the assembly line by focusing on launching and sequencing of (long) trucks at the assembly line.

Chapter 3

Literature

This chapter discusses literature related to this research. Section 3.1 discusses the state of the art of truck assembly lines and provides a framework for this research. Section 3.2 introduces relevant solutions that could be used to maximize the output of an assembly line with products with varying lengths. Section 3.3 discusses operations research techniques that we can use to design and formulate alternative solutions. To conclude, Section 3.4 formulates our choice of solution and evaluation methods to use based on this literature review.

3.1 State of the Art

In Section 3.1.1, we classify assembly lines based on their characteristics. Section 3.1.2 discusses related research on (truck) mixed-model assembly lines. Section 3.1.3 gives a detailed introduction on the assembly line sequencing problem, since we focus on this in our research.

3.1.1 Assembly Lines Classification

The main types of assembly lines are single-model, multi-model, and mixed-model, which are shown in Figure 3.1. The first assembly line was a single-model assembly line, introduced at the Ford Motor company, which only manufactured one product variant, the famous Model-T. Nowadays in the automotive industry, many product variants of a common based product are manufactured at the same assembly line. These variants are either assembled in batches with setup times, or in a random, mixed sequence without noticeable setup times, which are classified as multi-model and mixed-model assembly lines respectively (Kern et al., 2015). Mixed-model assembly lines are often used in assemble-to-order production systems and enable mass customisation (Boysen et al., 2009, 2010). Mass customisation is very common in the automotive industry (Alford et al., 2000), and SPZ similarly manufactures a wide variety of products in high volumes. Both assembly lines of SPZ are classified as mixed-model assembly lines. Because of flexible workers and machinery, it is possible to manufacture in lot sizes of one in an intermixed product sequence.

Assembly lines of large complex products, such as trucks, are often multi-manned and two-sided (Abdullah Make et al., 2017; Jawahar et al., 2014; Yilmaz & Yilmaz, 2019). This means that multiple operators work at the same work station on both sides of the line, either performing individual tasks or work together at heavy tasks.

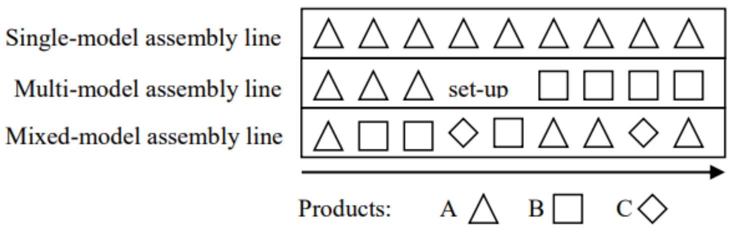


Figure 3.1: Main types assembly line (Kern et al., 2015)

3.1.2 Related Research on Mixed-Model Assembly Lines

In the literature on mixed-model assembly lines, two main problems can be distinguished: on the strategic and tactical level the assembly line balancing problem and on the operational level the assembly line sequencing problem. The goal of the assembly line balancing problem is to assign tasks with precedence constraints to stations in such a way that the cycle time is balanced over the stations (Boysen et al., 2021). Objectives for the balancing problem include minimization of the cycle time, the number of work stations, the costs, or some smoothness index (e.g., variance of work station cycle times). For an extensive review on two-sided assembly line balancing problems, we refer to Abdullah Make et al. (2017).

The objective of the assembly line sequencing problem is to determine an inter-mixed production sequence that either minimizes work overload or distributes the material requirements evenly (Boysen et al., 2009). Since this is in the scope of this research, Section 3.1.3 discusses this problem in more detail. Besides these problems, the operations research literature also discusses the part feeding of mixed-model assembly lines (Kilic & Durmusoglu, 2015), but this is out of scope for this thesis.

We give a few examples of existing research on truck mixed-model assembly lines. For overall improvement of the truck assembly line, Wirabhuana et al. (2008) use a computer simulation model to design and analyse improvement alternatives such as line balancing. Martignago et al. (2017) propose an integer linear program for the balancing problem that minimizes the total costs of a manual assembly line of large products, taking into account different operator skills levels. Shi et al. (2021) propose a genetic algorithm to solve a double objective assembly line balancing problem, which optimizes both the *production beat* (takt) and the smoothing factor. Rodríguez Parral and López Pérez (2020) discuss the effects and interactions of variables used in heavy truck assembly line sequencing models, based on multi-variable predictive regression models. They analyse that only the variable *installed capacity* significantly impacted all output variables: the number of finished pieces, the processing time, and the utilization. If the takt time was taken into account in the sequencing model, only the number of finished pieces and the processing time was significantly impacted.

Mixed-model assembly lines can be both paced or unpaced (Salehi et al., 2012). An unpaced line includes work stations that are decoupled by inventory buffers, whereas a paced line has coupled stations by material handling equipment such as a conveyor belt. Paced lines can be moving or synchronous. The first part of SPZ's assembly lines, the stop & go system, is synchronous, since the products stay in the stations during the cycle time and are synchronously moved to the succeeding station. The second part of the lines is moving, where a transportation system moves the products steadily from station to station. For a paced mixed-model assembly line, takt times are used to determine the pace of the line, and therefore the output rate. The cycle time, that refers to the processing time per work station, should be designed such that it aligns with the takt time, based on the customer demand (Dinesh et al., 2005). As a result, all production departments can manufacture the same amount each day, while the product mix can be based on the customer demand (Black, 2007; Suh et al., 1998). In automotive engineering, the paced mixed-model assembly line based on takt time is very common (Fang et al., 2013). Another benefit of the takt time approach is that the work in progress (WIP) can be reduced to a minimum, especially if this is integrated with other just-in-time (JIT) techniques (Ali & Deif, 2014). We give a few examples in literature that deal with paced moving assembly lines. Zhao et al. (2004) propose a line-balancing heuristic for such a paced moving mixed-model assembly line, which minimizes the total overload time. The overload time in a paced moving assembly line is the time that an operator needs outside of its own workstation to finish a task, which leads to stoppage time. Another way of dealing with this overload is the use of jolly operators, a flexible high-skilled workforce that supports the regular workforce, which is also the case at SPZ. Faccio et al. (2015) introduces a balancing and sequencing approach for a paced mixed-model assembly line, including these jolly operators.

To conclude, this thesis discusses a moving multi-manned two-sided mixed-model assembly line, where products have varying lengths. To our knowledge, this specific problem has not been discussed in literature yet. However, there is a lot of research available on the sequencing and balancing of such assembly lines. The contribution of this thesis is that we include large, complex products with varying lengths.

3.1.3 Assembly Line Sequencing Problem

The three well-known sequencing approaches in literature are mixed-model sequencing, car sequencing and level scheduling (Boysen et al., 2009). Mixed-model sequencing minimizes the sequence-dependent work overload explicitly based on operational characteristics of the assembly line. Car sequencing however, minimizes the work overload implicitly based on sequencing rules formulated as $H_0 : N_0$, which states that only H_0 products can have a specific trait among N_0 sequential products. Level scheduling aligns the sequence with the just-in-time (JIT) philosophy and focuses on levelling the part feeding process.

The assembly line sequencing problem and assembly line balancing problem can also be solved simultaneously, for example by a multi-objective artificial bee colony algorithm (Saif et al., 2014), or mixed-integer programming formulations (Sawik, 2004).

The sequencing problem that minimizes work overload is NP-hard (Tsai, 1995). So, metaheuristics are often used to solve this problem, such as robust simulated annealing (Cho et al., 2005), metaheuristics combined with reinforcement learning (Brammer et al., 2021), or a genetic algorithm, which is often applied in sequencing problems (Serkan Akgündüz & Tunali, 2010). Section 3.3.2 further explains metaheuristics and specifically the genetic algorithm. For a classification of sequencing literature we refer to the review of Boysen et al. (2009).

3.2 Relevant Solutions from Literature

This section discusses relevant research that possibly reduces the output loss caused by long vehicles on a mixed-model assembly line. Section 3.2.1 focuses on flexibility mechanisms in assembly lines: relaxing the fixed takt time constraint, using a flexible workforce and flexibly arranging assembly lines. Section 3.2.2 describes optimal ways of discrete measurements of the lengths of the trucks.

3.2.1 Flexibility in Assembly Lines

Svensson Harari et al. (2014) discuss three measures of flexibility of an assembly line: volume flexibility, mix and operation flexibility, and new product and removal of existing product flexibility, which they study at Scania Production Sweden and a comparable manufacturer. To flexibly increase the volume, useful mechanisms include having flexible personnel, decreasing the takt time, re-balancing the line, analysing the stations' length connected to the conveyor speed, and using paced assembly lines. Mechanisms to achieve product mix flexibility include having mutual standardized assembly processes, specific assembly areas for high workload products, and one assembly line for a variety of product models. Operation flexibility can be achieved by Andon systems, re-planning and re-scheduling of products, no sequencing rules and line stop options.

Relaxing the Fixed Takt Time Constraint

At SPZ, the output loss of long trucks cannot be compensated anymore, because there is a maximum of one truck per takt time. In literature, there are a few examples of flexible or variable takt times in assembly lines.

Huchzermeier et al. (2020) describe the *VarioTakt* approach at Fendt, a tractor manufacturer. In this approach, every model has its own specific takt time, which leads to a variable launching rate on the assembly line. Hence, operators have enough time for heavy workload products and experience a smooth workload division despite the model variations. Figure 3.2 shows how different models have their own takt time and thus their own launching rate on the assembly line. In this approach, the assembly line balancing problem is model-specific, and the sequencing of the products is less complex, since workload constraints are already taken into account in the takt times. However, a limitation of this approach is that the equipment and part feeding of every work station should be very flexible, since the tasks per work station are balanced individually for each model. For example, this could mean that axles for a 8x4 truck are assembled at another station than axles for a 4x2 truck, which requires independent axle supplies and equipment. To deal with this limitation, Mönch et al. (2020) introduce a heuristic to determine variable takt time groups, such that each product is divided into a group with a suitable takt time. They show that this group division results in higher labour efficiency.

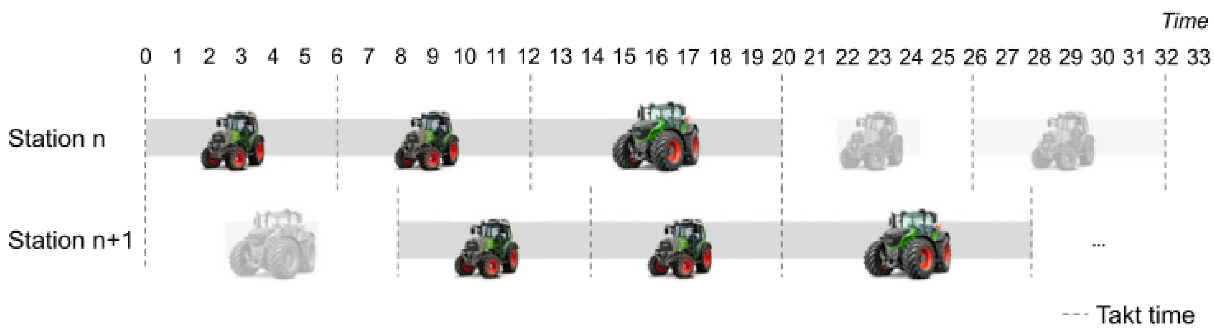


Figure 3.2: Illustration of VarioTakt at Fendt (Huchzermeier et al., 2020)

Mönch et al. (2021) formulate a mixed-integer programming model for the mixed-model assembly line balancing problem, based on the VarioTakt approach and taking operating work zones into account. They show that relocating operators to optimal work zones reduces the takt time significantly, and that the VarioTakt approach reduces the complexity of the mixed-model assembly line balancing problem in general.

Fattahi and Salehi (2009) introduce the concept of *variable rate launching*, where products are launched in various launching intervals on a conveyor belt with a constant speed. They propose a simulated annealing algorithm that determines a sequence and the launching intervals of all products in that sequence. The input and output of this heuristic is shown in Figure 3.3.

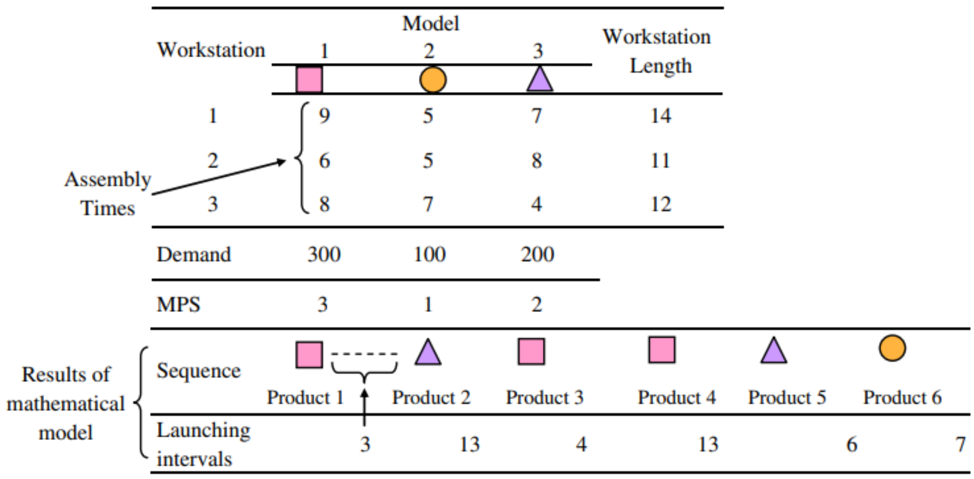


Figure 3.3: Illustration of variable rate launching sequencing algorithm (Fattahi & Salehi, 2009)

Zhang et al. (2021) propose to use dynamic takt times for a paced mixed-model assembly line. The dynamic takt time approach assumes that the fixed takt time is based on a bottleneck time that only occurs at certain stations for certain products, such that the dynamic takt time is set to the maximum of all processing times per takt time. Figure 3.4 illustrates this and shows that the production rate is dynamically changing based on the highest processing time in that status. In a fixed takt time approach, every status would take 10 minutes, which results in much more idle time. This approach is only applicable to a synchronous assembly line, or in a complex material handling system where products can independently move with different speed.

Status	Workstation	Workstation	Workstation	Workstation	Workstation	Idle time	Starting time	End time
Status 1	$t_{A1}=6$					$t_{s1}^{idle}=0$	$t_{s1}^{start}=0$	$t_{s1}^{end}=6$
Status 2	$t_{B1}=1$	$t_{A2}=8$				$t_{s2}^{idle}=7$	$t_{s2}^{start}=6$	$t_{s2}^{end}=14$
Status 3	$t_{C1}=8$	$t_{B2}=2$	$t_{A3}=6$			$t_{s3}^{idle}=8$	$t_{s3}^{start}=14$	$t_{s3}^{end}=22$
Status 4	$t_{A1}=6$	$t_{C2}=5$	$t_{B3}=6$	$t_{A4}=3$		$t_{s4}^{idle}=4$	$t_{s4}^{start}=22$	$t_{s4}^{end}=28$
Status 5	$t_{B1}=1$	$t_{A2}=8$	$t_{C3}=3$	$t_{B4}=2$	$t_{A5}=4$	$t_{s5}^{idle}=22$	$t_{s5}^{start}=28$	$t_{s5}^{end}=36$
Status 6	$t_{C1}=8$	$t_{B2}=2$	$t_{A3}=6$	$t_{C4}=10$	$t_{B5}=7$	$t_{s6}^{idle}=17$	$t_{s6}^{start}=36$	$t_{s6}^{end}=46$
Status 7	$t_{A1}=6$	$t_{C2}=5$	$t_{B3}=6$	$t_{A4}=3$	$t_{C5}=4$	$t_{s7}^{idle}=6$	$t_{s7}^{start}=46$	$t_{s7}^{end}=52$
Status 8	$t_{B1}=1$	$t_{A2}=8$	$t_{C3}=3$	$t_{B4}=2$	$t_{A5}=4$	$t_{s8}^{idle}=22$	$t_{s8}^{start}=52$	$t_{s8}^{end}=60$

Figure 3.4: Illustration of dynamic takt times (Zhang et al., 2021)

Flexible Workforce

A flexible workforce presents efficiency opportunities for manual assembly lines, for example with the method OFRO (organizing flexible, rotating operators) (Downey & Leonard, 2007). If operators are allowed to move to open stations if their own tasks are finished, the delay of the line can be minimized. It is also possible that operators can move a predefined distance out of their stations, as shown in Figure 3.5. Mohseni-Darabi et al. (2021) propose a bi-objective mathematical model for the paced multi-manned mixed-model assembly line sequencing problem, taking these operators into account. The jolly operators proposed by Faccio et al. (2015) could also create flexibility (see Section 3.1.2).

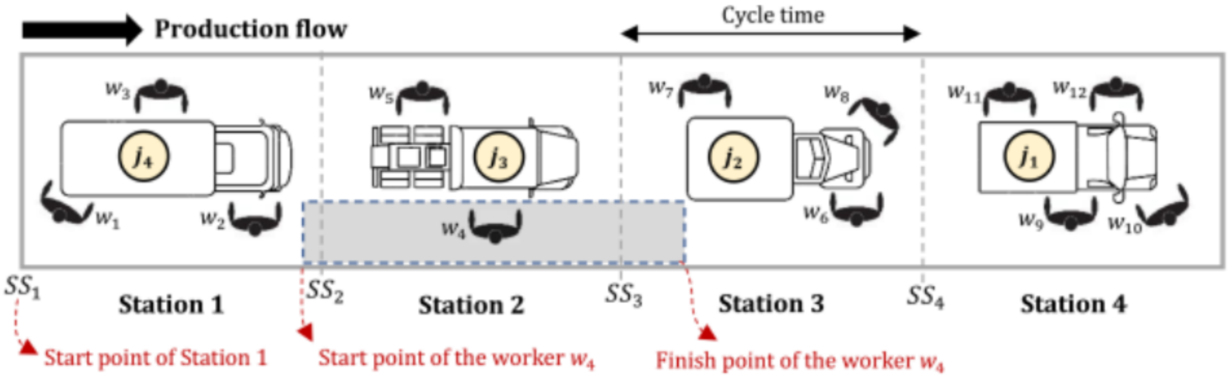


Figure 3.5: Illustration of moving operators (Mohseni-Darabi et al., 2021)

Flexible Arrangement of Assembly Lines

Although mixed-model assembly lines are designed such that they can manufacture all possible models, the planning department still has to schedule products on certain assembly lines. Prombanpong et al. (2010) determine the fixed rate launching of multiple mixed-model assembly lines, and the car sequence per line using the fixed rate launching algorithm. Hemig et al. (2014) propose an integrated dynamic programming approach with a heuristic to assign products to heterogeneous assembly lines and schedule staff on these lines.

Kampker et al. (2020) propose a hybrid assembly structure instead of a mixed-model assembly line. These hybrid assembly structures are segmented into decoupled flexible segments for variant-specific products, and line segments with variant non-specific products. Each line segment has an individual predetermined sequence, such that the JIT supply can still take place and certain products can move through the buffer of the flexible segment directly.

3.2.2 Optimal Discrete Length Measurement

The optimal placement of the light sensors that minimizes the unnecessary extra distance between consecutive trucks can be compared to a facility location problem. If there is a discrete set of possible locations for the light sensors, the problem is very similar to a p-median model. This model locates p facilities to minimize the demand-weighted total distance between demands and the nearest facility (Daskin, 2008). At SPZ, the light sensors can be compared with the facilities and the end of the truck corresponds to the location of the demand, where the distance to the nearest facility (the first sensor that does not detect the truck) should be minimized. Daskin (2008) formulates the p-median model as follows:

$$\min \sum_{j\in J}\sum_{i\in I} h_i d_{ij} x_{ij} \tag{3.1}$$

$$\text{s.t.} \sum_{j\in J} x_{ij} = 1, \quad \forall i \in I \tag{3.2}$$

$$x_{ij} - y_j \leq 0, \quad \forall i \in I, \quad \forall j \in J \tag{3.3}$$

$$\sum_{j\in J} y_j = p \tag{3.4}$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in I, \quad \forall j \in J \tag{3.5}$$

$$y_j \in \{0,1\}, \quad \forall j \in J \tag{3.6}$$

In this formulation, p facilities have to be located at the set of locations J. If location j is chosen, then binary decision variable $y_j = 1$. The set of demand points is denoted by I and the demand weight of location i is notated by h_i. The assignment of demand point i to location j is shown by the binary decision variable x_{ij}. (3.1) gives the objective function that minimizes the demand-weighted total distance between the demand points and their nearest location. Constraints (3.2) and (3.3) ensure that all demand points are assigned to an opened location. Constraint (3.4) ensures that p facilities are opened, and constraints (3.5) and (3.6) are integrality constraints. (3.5) can be relaxed to (3.7), since it will automatically be assigned to the closest open site in all feasible solutions (Daskin, 2008).

$$0 \leq x_{ij} \leq 1, \quad \forall i \in I, \forall j \in J \tag{3.7}$$

If there is no discrete set of locations for the light sensors, an analytical or continuous location model could also be applied. An analytical model assumes that the demand is distributed according to some known distribution, while continuous models assume that the there is a discrete set of demand points (Daskin, 2008).

3.3 Techniques for Optimization of Assembly Lines

Vehicle Assembly Lines are optimized using a heuristic approach, mathematical modelling approach, computer simulation approach, search algorithm, or a combination of simulation and search algorithm (Rane et al., 2015). The heuristic approach refers to the use of several Lean techniques, such as assembly line balancing, use of the Kanban pull system and reduced WIP. Discrete-event simulation (DES) is a commonly used simulation approach, which uses generated events to model the real world based on logical processes (Cassandras & Lafortune, 2008). DES models the real-world to measure the performance of solutions, but does not improve on the solutions themselves. Therefore, a combination of DES with a search algorithm, such as simulated annealing or a genetic algorithm, is a useful method to optimize vehicle assembly lines. This section discusses both techniques: Section 3.3.1 discussed *simulation-based optimization* and Section 3.3.2 discusses metaheuristics. Section 3.3.3 introduces a combination of simulation-optimisation and heuristics: *simheuristics*.

3.3.1 Simulation-based Optimization

Optimizing a process using computer simulation is also called simulation-based optimization (short: simulation-optimization), which is frequently used if an objective function is not easily expressed as an analytical function. Then, computer simulation helps to evaluate a certain set of parameters and can be combined with meta-heuristics to find a close-to-optimal solution. Simulation-optimization problems can be classified based on their feasible solution region (Hong & Nelson, 2009). If there is a small number of solutions, all solutions can be simulated to find the best solution, which is called a ranking-and-selection problem. If the solution space consists of a vector of continuous decision variables, it is called a continuous problem, which is often solved with a stochastic optimization algorithm. If the problem has a large solution space consisting of discrete and integer order decision variables, it is called a discrete problem, which is solved by random search algorithms, given that simulation models are often viewed as black boxes. Wang and Shi (2013) classify the simulation optimization models as shown in Figure 3.6.

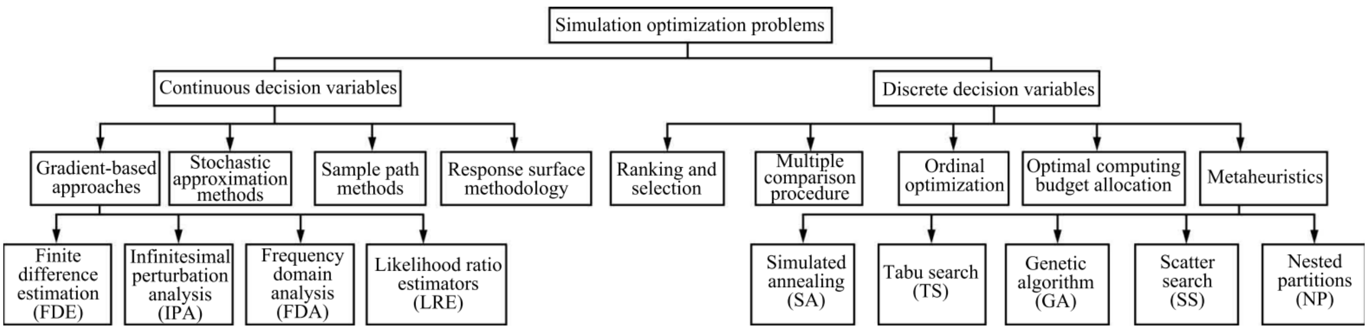


Figure 3.6: Classification of Simulation-Optimization models (Wang & Shi, 2013)

In recent literature, simulation is often used as a technique to optimize mixed-model assembly lines. It can be used to quickly evaluate operational decisions for an assembly line, or to support the balancing and sequencing problem (Tiacci, 2012). Lv et al. (2019) use a Monte-Carlo simulation that includes randomised processing times to determine bottleneck stations in a mixed-model assembly line. Biele and Manch (2015) propose a variable neighbourhood search for the mixed-model assembly line balancing problem, which results are discussed and analysed using a stochastic simulation model. Next to optimizing an assembly line, simulation can also be used for re-engineering of an assembly line. For example, Wirabhuana et al. (2008) use a simulation approach to re-engineer a truck assembly line, by modelling four different solutions, such as line balancing or parallel operation.

3.3.2 Metaheuristics

Local search heuristics tend to get stuck in local optima, such that the optimal solution cannot be found. Therefore, metaheuristics are designed to combine exploration (exploring other solution areas) and exploitation (find an optimal solution in one area) (Abdel-Basset et al., 2018). For sequencing problems, which have a very large solution space, the genetic algorithm is a well-known metaheuristic, since it explores multiple solutions in each iteration (Serkan Akgündüz & Tunali, 2010). The genetic algorithm is the most widely known evolutionary algorithm, which is based on Darwin's theory of evolution (Eiben & Smith, 2015). A pseudo-code of an evolutionary algorithm is given in Figure 3.7.

```
BEGIN
  INITIALISE population with random candidate solutions;
  EVALUATE each candidate;
  REPEAT UNTIL ( TERMINATION CONDITION is satisfied ) DO
    1 SELECT parents;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
    4 EVALUATE new candidates;
    5 SELECT individuals for the next generation;
  OD
END
```

Figure 3.7: Pseudo-code of evolutionary algorithm (Eiben & Smith, 2015)

The sequencing problem requires a permutation representation of the solutions. For this representation, commonly used mutations are *swap*, *insert*, *scramble* and *inversion* mutations. In these mutations, two positions are swapped (see Figure 3.8), an item is moved to another position, a (subset of) the sequence is randomly shuffled and a subset within the sequence is inverted (see Figure 3.9), respectively.

Figure 3.8: Swap mutation illustration (Eiben & Smith, 2015)

Figure 3.9: Inversion illustration (Eiben & Smith, 2015)

To generate new solutions (*offspring*), parent solutions are recombined (*mated*) using crossovers. Recombination methods that are often used with permutation represented solutions are: *partially mapped*, *edge*, *order* and *cycle* crossover. The partially mapped crossover (PMX) was proposed by Goldberg and Lingle (1985) for a well-known sequencing problem, the *travelling salesman problem.* The PMX swaps the positions of a subset of the sequence as far as this is possible, of which Figure 3.10 shows an example. First, a subset of the sequence is copied from parent 1 into the offspring. Then, the same subset of parent 2 is placed in the positions of the same genes in parent 1. In our example, this means that 8, which maps to the position of 4 in parent 1, is placed in the last position in the offspring, which is the position of 4 in parent 2. After this, the other genes from parent 2 are placed in their old position in the offspring. The edge crossover creates offspring that preserves edges (adjacent items) from the parents. The order crossover copies a subset from one parent to the offspring, and uses the relative order of the other parent to complete the offspring, of which Figure 3.11 gives an illustration. In this example, a random chosen subset from parent 1 is copied into the offspring first. Then, starting from the end of the subset of parent 2, the non-occurring items of parent 2 are placed in the offspring in their relative order, wrapping around at the end of the sequence. Finally, the cycle crossover preserves information on the absolute position of the items, rather than focusing on the order.

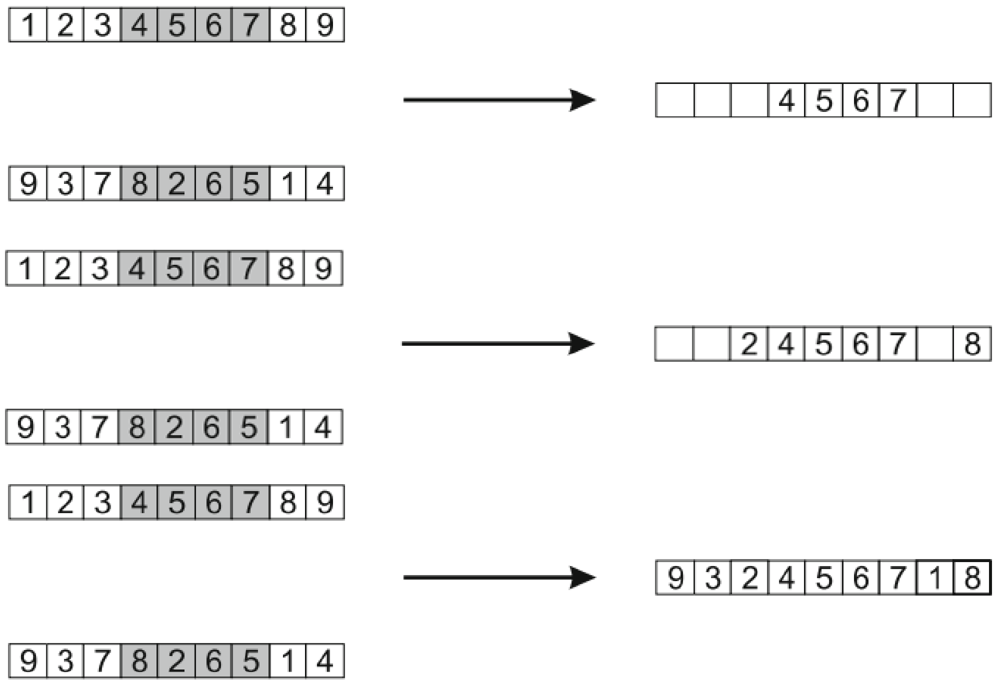


Figure 3.10: Partially mapped crossover Illustration (Eiben & Smith, 2015)

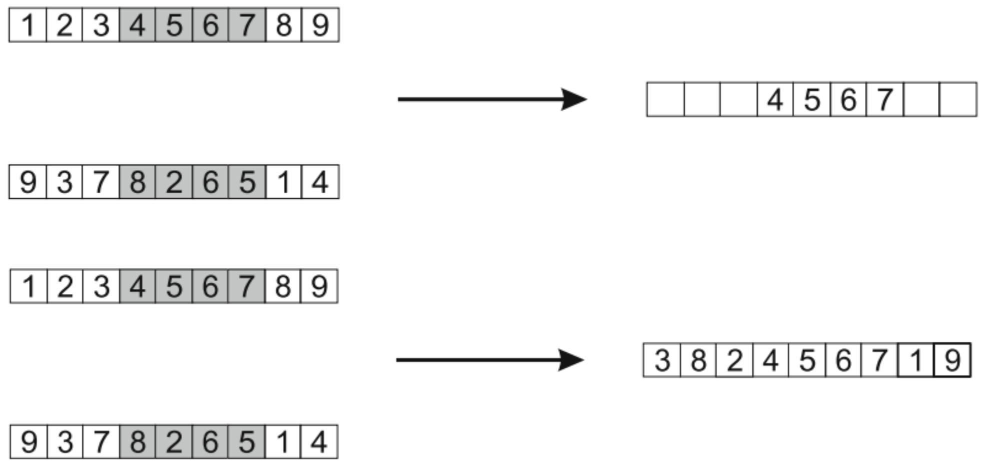


Figure 3.11: Order crossover illustration (Eiben & Smith, 2015)

For mixed-model sequencing, genetic algorithms are often applied with order crossover as mating process and inversion mutation, as shown in the review of Serkan Akgündüz and Tunali (2010). To select parents to mate, both roulette wheel selection and rank selection are used, which select a parent based on their *fitness value* (objective value) and rank, respectively. Many applications of a genetic algorithm in sequencing problems involve elitism, which means that the best individuals are preserved for the next generation. Parameter tuning is often used, which means that the parameters are fixed upfront based on experiments. An example of a genetic algorithm approach for a mixed-model assembly line that aims to minimize the number of lines stoppages, is given by Xiaobo and Ohno (2000). They determine the fitness value of a solution with a complex iterative calculation, to include stoppage and idle time correctly. They also prove that their problem is NP-hard in the strong sense, and come up with a simple heuristic to come to a good solution.

3.3.3 Simheuristics

Simulation-optimization can also be combined with heuristics, for which Figure 3.12 shows a procedure by Juan et al. (2015). For complex combinatorial optimization problems (COPs) with a complex stochastic simulation model that cannot evaluate all solutions, the simulation model can be combined with a model based on metaheuristics. In this way, a simplified, deterministic model can generate new solutions fast, of which only the promising solutions are measured by the stochastic model.

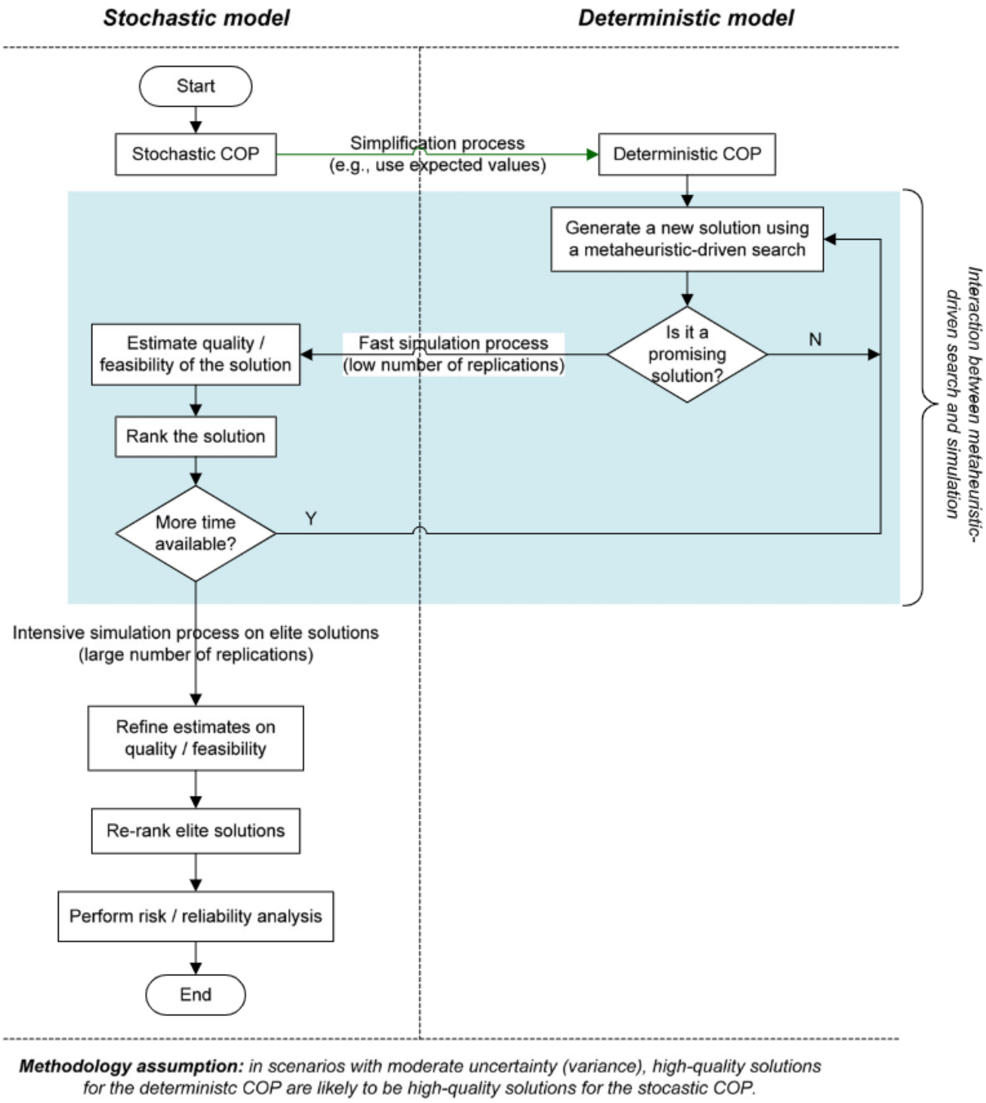


Figure 3.12: Methodology of simheuristics (Juan et al., 2015)

A limited literature review in Scopus and Web of Science using the key words *simheuristics* and *sequencing* showed that the mixed-model sequencing problem is not solved yet using simheuristics. However, the parallel flowshop scheduling problem is solved using simheuristics (Hatami et al., 2018). Vanguri et al. (2006) also solve the sequencing in a flow shop, without simheuristics but using a combination of simulation optimization with evolutionary methods. Every sequence generated by a genetic algorithm is evaluated by a DES.

3.4 Conclusion

To conclude, we classify the assembly line at SPZ as a paced moving, multi-manned, two-sided, mixed-model assembly line, with trucks with varying lengths. Based on this literature review, we propose to use a P-median location model for the positioning of the light sensors to measure the lengths of the trucks. To maximize the output of the mixed-model assembly line at SPZ, we propose to use variable rate launching in combination with mixed-model sequencing. We use a genetic algorithm to generate good mixed-model sequences and a DES model to evaluate alternative solutions.

Chapter 4

Solution Design

Based on our literature review, this chapter designs solution approaches for the core problems defined in Chapter 1: the number of and positions of the light sensors; the launching of the trucks on the carrier line; and the sequencing of the trucks. Section 4.1 formulates a linear program based on the facility location problem for the number and positions of the light sensors. Section 4.2 describes two scenarios that we develop based on literature for the launching of the trucks. Section 4.3 formulates two approaches of our designed genetic algorithm that generates near-optimal sequences of trucks.

4.1 Light Sensor Location Model

We formulate a discrete measuring location model for the light sensor positions to measure the truck lengths. This model is based on the P-median location model, since it has many similarities, as shown in Figure 4.1.

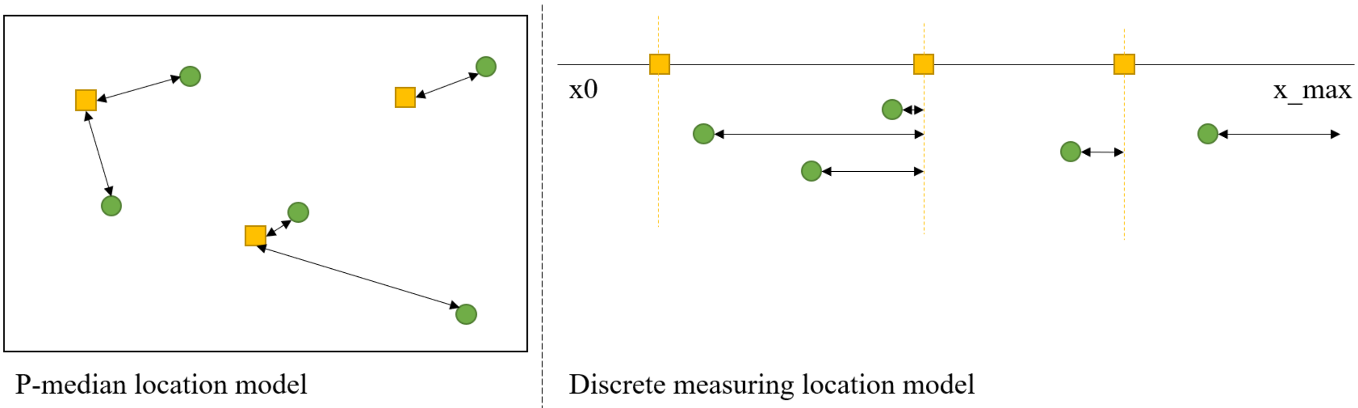


Figure 4.1: Comparison P-median location model and discrete measuring location model

The discrete measuring location model is formulated as follows:

Sets:
I = set of trucks, index i
J = set of location options for sensors (in mm), incrementally, index j

Parameters:
l_i = length of truck i

$$d_{ij} = \begin{cases} \text{distance from the end point of truck } i \text{ to sensor location } j, & \text{if } l_i < j \\ \text{relatively high number,} & \text{if } l_i >= j \end{cases}$$

p = number of sensors that can be placed

Decision Variables:

$$x_{ij} = \begin{cases} 1, & \text{if truck } i \text{ is given the driving distance of sensor location } j \\ 0, & \text{otherwise} \end{cases}$$

$$y_j = \begin{cases} 1, & \text{if a sensor is placed a location } j \\ 0, & \text{otherwise} \end{cases}$$

Objective Function & Constraints:

$$\min \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \tag{4.1}$$

$$\text{s.t.} \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I \tag{4.2}$$

$$x_{ij} - y_j \leq 0, \quad \forall i \in I, \quad \forall j \in J \tag{4.3}$$

$$\sum_{j \in J} y_j = p \tag{4.4}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in I, \quad \forall j \in J \tag{4.5}$$

$$y_j \in \{0, 1\}, \quad \forall j \in J \tag{4.6}$$

This model minimizes the extra distances between the truck lengths and the distance that is measured by the chosen positions of the sensors, see (4.1), in order to minimize the unnecessary output loss. Constraints (4.2) ensure that every truck is assigned to exactly one sensor location, the distances ensure that trucks are only assigned to a sensor that the truck does not cross. For this, the assumption holds that all truck lengths do not exceed the maximum sensor location. Constraints (4.3) ensure that if a truck length is measured by a sensor at a location, a sensor should also be placed at that location, and constraint (4.4) ensures that exactly p sensors are placed. Constraints (4.5) and (4.6) are binary integrality constraints, where constraints (4.5) can be relaxed to constraints (4.7). This is possible, because y_j is a binary variable, and the distances linearly increase with the sensor locations, so each truck will still be assigned to one sensor location.

$$0 \leq x_{ij} \leq 1, \quad \forall i \in I, \forall j \in J \tag{4.7}$$

We can use this model to determine the optimal positions for the sensors to minimize the extra measured distance, in order to maximize the daily output of SPZ. By changing the parameter p, the management of the engineering department can decide which number of sensors leads to a decrease in the extra measured lengths and is still worth the investment of extra sensors.

4.2 Truck Launching Strategies

The literature review showed multiple promising solutions to create flexibility in assembly lines. The concept of *variable rate launching* seems the most promising for SPZ to deal with variable length trucks on an assembly line with a constant speed, but also requires a few organisational changes. Therefore, we develop two alternative launching strategies for SPZ: variable rate launching, and *triple takt time*.

In the variable rate launching strategy, every truck is launched on the carrier line as soon as it is finished at the engine station (Station 29) and there is a safe space on the carrier line. As a result, shorter trucks can compensate for the idle time that is caused by long trucks, which we expect to work best if stations are open. We expect this strategy to also compensate idle time with other causes, such as stops due to long task times. However, variable rate launching will be relatively difficult to implement at SPZ, because management is used to organizing the entire plant based on the fixed takt times and fixed rate launching (except for long trucks). Therefore, we also analyse a strategy that is more in line with the Lean philosophy at SPZ, the triple takt time.

In the triple takt time strategy, developed by SPZ's development engineer H. Oolman, every three consecutive trucks can at most occupy the carrier line for the length of three consecutive stations (36 m). Then, we view a long truck as the first truck of this sequence which occupies more space on the line, which the second and third truck are allowed to compensate. These shorter following trucks cannot start earlier than the delayed start that is caused by a long truck, so this strategy only compensates idle time caused by long trucks. This strategy requires the same technical adjustments of the assembly line as the variable rate launching strategy. However, the concept of fixed takt times remains in the triple takt time strategy, so this strategy could be easier to implement at SPZ.

We expect that both alternative solutions will increase the daily output of the assembly line at SPZ compared to the current launching strategy.

4.3 Sequencing

Scania Sweden implemented car sequencing for all final assembly units, because the sequencing rules are easy to use in practice. In this research however, we want to include the length, and thus the occupancy of a truck in a station in the sequencing decision. Since both car sequencing and level scheduling do not take this high level of detail into account, and the logistical perspective is out of scope for this research, we use mixed-model sequencing.

Our mixed-model sequencing problem has the following characteristics. The station boundaries at SPZ are open to some extent, the reaction to imminent work overload is line stoppage and the processing times are stochastic. We assume that work cannot happen concurrently, which means that operators from different stations cannot work at the truck simultaneously. SPZ does not consider setup times nor costs, and there are no parallel stations. There is a fixed number of stations, and although the station boundaries vary, we assume that the stations are homogeneous. The current launching discipline is fixed rate launching (except for long trucks), but we also want to experiment with variable rate launching. We take a return walking time for the operators into account, and the line layout consists of serial stations. The objective of this problem is to maximize the output, which implicitly minimizes the duration of line stoppages, work overload and total idle time. Based on the classification of Boysen et al. (2009), we classify our mixed-model sequencing problem as $[open; stop; p^{sto}|n; vl; fin|Co(wo; idle; stop)]$.

To solve the mixed-model sequencing problem, we use a genetic algorithm, in which the quality of many sequences have to be evaluated. However, the evaluation of the sequences requires high computational effort, since we cannot express the objective function (the total output given a certain sequence) in an analytical function. So, to find a near-optimal solution for our mixed-model sequencing problem, we have to make a trade-off between the optimisation effort and the quality evaluation effort. Therefore, we design three approaches: the *simplified evaluation approach (SE)*, which focuses on finding the optimal solution for the problem with a fast, simplified evaluation method, the *discrete-event simulation approach (DES)*, which focuses on evaluating the solutions with a complex DES model and the *simheuristics approach (SH)* which combines both approaches. Section 4.3.1 discusses the SE approach, Section 4.3.2 presents the DES approach and Section 4.3.3 introduces the SH approach.

4.3.1 Simplified Evaluation Approach

The SE approach uses a genetic algorithm to optimise the sequencing of the mixed-model assembly line, with a deterministic, simplified and thus fast evaluation function, such that most of the available computation time can be spent on optimisation. The genetic algorithm (GA) for the mixed-model sequencing of trucks includes both the processing times and lengths of the trucks. We discuss all elements of the GA individually, and then present the pseudo-code.

For the genetic representation, we use permutation encoding, since each solution consists of a possible sequence of the trucks. For example, a solution could be represented by: [2, 3, 1, 4] or [1, 2, 3, 4], where the numbers represent the product individual numbers of the trucks. We generate an initial generation using sequences where the long trucks are spread evenly, because sequences with clustered long trucks tend to give bad results. we use the fitness values relative to the worst fitness value for the selection procedure, because the makespans of the sequences tend to be close together, and we want to distinguish between the fitness values. In the common selection approach, *roulette wheel selection*, the probability of selecting a sequence is based on its fitness value. If these values are high and close together, as in our research, this leads to similar selection probabilities for all sequences. In another common selection approach, *rank selection*, the generation is sorted according to their fitness value, and then given a rank. The selection probability of each solution is then determined by dividing its rank by the sum of all ranks: $prob_{selection} = r_{\pi(n)} / \sum_{n \in C} r_{\pi(n)}$ (Kumar & Jyotishree, 2012). In this approach, we cannot distinguish whether fitness values of the best sequences are close together or not. Therefore, we decided to use a combination of these selection approaches. After creating the children, we stochastically select the offspring for the new generation from both the children and the parents based on their fitness values. We also apply elitism, by replacing the five worst offspring from generation i by the five best parents from generation $i-1$. In this way, we are able to always include the best sequences in our population size. Although adaptive parameter control could lead to slightly better results, the implementation of parameter tuning is easier in practice. Therefore, we decide to use parameter tuning, similar to many studies that applied a GA in practice (Serkan Akgündüz & Tunali, 2010). We combine all these elements in the following GA:

Algorithm 1 Pseudo-Code Genetic Algorithm Mixed Model Sequencing

```
Create_Initial_Sequences;                                          [Spread long trucks evenly]
Evaluate_Sequences(Sequences_Init);                                [Using formulas (4.8) - (4.20)]
Sequences_Curr := Sequences_Init;
Results_Curr := Results_Init;
while not termination do
   Calculate_Selection_Probabilities(Sequences_Curr);              [Using rank selection]
   Sequences_New = ∅;
   for i in size_population do
      Parents := Select_Parents(Probabilities_Selection);
      Child := CrossOver(Parents);                                 [Both parents have equal probabilities]
      if randomnr ≤ prob_mutation then
         Mutate(Child);
      end if
      Sequences_New + = Child;
   end for
   Elitism(Sequences_New, Sequences_Curr, 5);                      [Swap worst offspring with best parents]
   Sequences_Curr := Sequences_New;
   Results_Curr := Evaluate_Sequences(Sequences_Curr);
end while
return Sequences_New, Results_Curr
```

For the evaluation of the quality of the sequences, we develop a simplified, deterministic simulation model. The objective of our mixed-model sequencing problem is to maximize the number of trucks that can be assembled. Since the input of our problem is a finite data period set with trucks, the fitness value in this approach represents an estimate of the makespan, which should be minimized. To include the lengths of the trucks, we define the concept of *occupation time*, which represents the time that a truck takes to fully occupy a station. This section introduces our simplified, deterministic simulation model. We make the following assumptions to ensure that we can quickly calculate an estimate of the makespan of a sequence:

1. The assembly line is only stopped because of Castor2 stations that could not finish the tasks of a truck in time (because the truck already left the stations).We exclude stoppages with different causes.
2. A new truck is available at the start of Castor2 exactly one takt time after its preceding truck entered the first station, independent from any stoppages at Castor1 and/or Castor2.
3. The length of the truck is measured continuously and we do not take a discrete set of carrier distances in account.
4. The stations are closed and a truck enters a station at a takt time after its start time of the previous station.
5. The processing times are deterministic and based on the average processing times. The walking time of operators is also deterministic and independent of where an operator ends in the station.
6. We add the same set of 20 (number of stations - 1) dummy trucks before and after the sequence, to include a warm-up and cool-down period.

We use the following notation:

S = Set of stations, index s

C = Set of all trucks, index c

D = Set of dummy trucks, $D \subset C$

$\pi(n)$ = The nth unit in sequence $\pi = \{\pi(1), ..., \pi(|S|)\}$

l_c = Length of truck c

t_c^s = Processing time of truck c at Station s

o_c = Occupation time of truck c for every station because of its length

$takt$ = Takt time

v = Line speed

$w =$ Walking time for operator
$ist_c^s =$ Initial start time of truck c at Station s (without considering stoppages)
$st_c^s =$ Actual start time of truck c at Station s
$et_c^s =$ End time of truck c at Station s
$ft_c^s =$ Time that truck c enters Station s
$gt_c^s =$ Time that truck c leaves Station s
$P =$ List of stops, index p
$b^p =$ Start time of stop p
$e^p =$ End time of stop p
$m =$ Makespan of sequence

We iteratively calculate all start, end, enter and leave times of all trucks per station, in order of the sequence π and of the stations (truck 1 on Station 1, truck 1 on Station 2, truck 2 on Station 1, etc.), using the following formulas:

$$o_c = l_c/v, \quad \forall c \in C \tag{4.8}$$
$$ft^s_{\pi(n)} = gt^{s-1}_{\pi(n)}, \quad where s > 1 \tag{4.9}$$
$$ft^1_{\pi(n)} = 0 \tag{4.10}$$
$$ist^s_{\pi(n)} = \max(et^s_{\pi(n-1)} + w; et^{s-1}_{\pi(n)}; ft^{\pi(n)}_s), \quad \text{where } n > 1, s > 1 \tag{4.11}$$
$$ist^1_{\pi(n)} = \max(et^s_{\pi(n-1)} + w; ft^s_{\pi(n-1)} + takt; ft^s_{\pi(n-1)} + o_{\pi(n)}), \quad \text{where } n > 1 \tag{4.12}$$
$$ist^1_{\pi(1)} = 0 \tag{4.13}$$
$$ist^s_{\pi(1)} = et^{s-1}_{\pi(1)}, \quad \text{where } s > 1 \tag{4.14}$$
$$st^s_{\pi(n)} = \max(ist^s_{\pi(n)}; \max_{p \in P, \text{ where } b^p < ist^s_{\pi(n)}} (e^p)) \tag{4.15}$$
$$et^s_{\pi(n)} = st^s_{\pi(n)} + t^s_{\pi(n)} \tag{4.16}$$
$$gt^s_{\pi(n)} = ft^s_{\pi(n)} + takt + \max_{p \in P, \text{ where } b^p >= ft^s_{\pi(n)}} (\min(e^p, gt^s_{\pi(n)}) - b^p) \tag{4.17}$$
$$b^p = gt^s_{\pi(n)}, \quad \text{if } (et^s_{\pi(n)} > gt^s_{\pi(n)}) \tag{4.18}$$
$$e^p = et^s_{\pi(n)}, \quad \text{if } (et^s_{\pi(n)} > gt^s_{\pi(n)}) \tag{4.19}$$
$$m = et^S_{\pi(C-D)} - st^1_{\pi(D+1)} \tag{4.20}$$

The initial start time of truck c at station s is the maximum (see (4.11)) of these three variables:

1. the end time of truck $c - 1$ at station s, including walking time for the operators,
2. the time that truck c is finished at station $s - 1$ (for $s = 1$, this is based on the takt times, see (4.12)), and
3. the time that truck c enters the boundaries of station s (for $s = 1$, this is based on when the last truck left the station physically, based on its occupation time, see (4.12)). This enter time is based on when the truck left its last station, or 0 for the first station, see (4.9) and (4.10).

The first truck at the first station starts at time 0 (see (4.13)), and the start time of the first truck at the other stations only depends on its end time at the last station (see (4.14)). If there is a stoppage that starts before the initial start time of truck c at station s, the truck can only start after the end time of this stoppage, as shown in (4.15). The end time of truck c at station s only depends on its start time and processing time ((4.16)). If the end time of a truck exceeds the leave time, a stop p is planned given the formulas (4.18) and (4.19). The truck leaves its station after a takt time of its enter time, extended by the longest stoppage that started after the enter time (see (4.17)). If a stoppage delays the leave time of a truck, but this is determined later in the iteration, we still extend the leave time by the length of this stoppage. After all iterations, the makespan is determined by the difference of the end time of the last truck of the sequence and the start time of the first truck of the sequence, excluding the dummy trucks, see (4.20).

4.3.2 Discrete-Event Simulation Approach

In the DES approach, we use a complex DES model to evaluate the quality of the sequences. We also use this model, which Chapter 5 further explains, to evaluate all alternative solutions. Since we designed the DES model in the Plant Simulation software of Siemens, we use the built-in GA functionality of this software. The GA has the same elements as the SE approach, except for some limitations of the software. In Plant Simulation, it is not possible to add initial sequences for the first generation, so these sequences are generated randomly. For the parent selection, rank selection is not available in the software, the fitness values are either referenced absolute, or relative to the worst value, in order to create more difference between the best and worst sequences. We apply elitism by cloning the best solution from generation i to generation $i+1$. Since the outcomes of the DES model are stochastic, the GA takes three observations of each sequence to determine the fitness value, for which we refer to Appendix A.

4.3.3 Simheuristics Approach

In the SH approach, we run one iteration of the simheuristics methodology by Juan et al. (2015), further described in Section 3.3.3. This is a combination of the aforementioned approaches, since it applies a fast, deterministic solution evaluation in a DES environment. First, we run a GA in our DES model, where each evaluation consists of a deterministic run with only one batch, which minimizes the makespan. For this GA, we use the same elements as in the DES approach. Then, we evaluate the resulting 100 best sequences in the stochastic DES model with multiple run days and observations, to find a near-optimal sequence.

4.4 Conclusion

To conclude, we designed alternative solutions for all three improvement possibilities we discovered in Chapter 2. We formulated an exact LP for the optimal positions of the light sensors. For the truck launching, we designed two strategies, the triple takt strategy that only compensates long trucks, and the variable rate launching strategy that launches all trucks as soon as possible. Lastly, we developed three approaches to generate near-optimal sequencing for the mixed-model sequencing problem using a genetic algorithm.

Chapter 5

Solution Evaluation

This chapter discusses our evaluation methodology of the designed alternative solutions. Section 5.1 discusses our discrete-event simulation model. Section 5.2 describes our experiment design, of which we discuss the results in Chapter 6.

5.1 Discrete-Event Simulation

This section discusses the design of the discrete-event simulation (DES) model that evaluates the GA-generated solutions, the designed launching strategies and the determined light sensor positions. Section 5.1.1 gives a description of the model, including its objective, scope and assumptions. Section 5.1.2 discusses the conceptual modelling of some complex processes in our DES. Section 5.1.3 shows the implemented final model and describes its validation and verification.

5.1.1 Model Description

The goal of this simulation is to evaluate the designed alternative solutions, which all have the objective to maximize the daily output, and to compare these solutions with the benchmark of the current situation. Maximizing the output corresponds with minimizing the stoppage times of the assembly line, so we use the key performance indicators (KPIs) *average daily production rate* and *average stoppage time percentage per line part*. Since the efficiency of the assembly line is highly influenced by its operators, we also include the KPI *average idle time percentage*. Since some solutions are based on the launching rate of the trucks on an assembly line, the goal of the simulation also includes giving visual insight in the flow and spacing of the trucks on the assembly line. The scope of our model only includes the Castor assembly line, since this is the high volume line that includes more variation in the trucks. Therefore, this is the most interesting line to test various sequencing approaches and to gain the most improvement. Since we focus on the paced moving carrier line, the first part of the assembly line (Castor1) is out of scope for our solutions. We do include the last stations of Castor1 in our DES model to simulate a realistic arrival rate. The balancing of the tasks of the stations and the station length are out of scope.

The variables that we want to change in our model are:

1. The order mix (trucks with their lengths, processing times per station and sequence)
2. Variance-to-mean ratio for the processing times
3. Walk time for operators from the end of a station to the start
4. Left and right boundaries of the stations (when operators can start and stop with working on a truck)
5. Number of light sensors and their positions
6. Launching method of trucks on the carrier line

The model is divided into four parts, such that everything can be clearly shown without over-complicating the model, such that it can be used by SPZ's process engineering department (see Figure 5.1). The most complex part of the model is Part 2, which includes the transition of the synchronously moving hanging conveyor to the constant driving carriers. In Part 1, the trucks are created and synchronously moved through the stations. Since Castor1 is mostly out of scope for this research, we only included Station 26 until 27A of this part, to recreate the flow of the arrivals at Castor2. Part 3 and 4 show the flow of the trucks on the carriers, to visually show the driving distances between the carriers. Table C.3 in Appendix C shows the input of our DES model, which is based on similar analysis as in Section 2.3, but for the entire period of the current generation trucks until March, 2022.

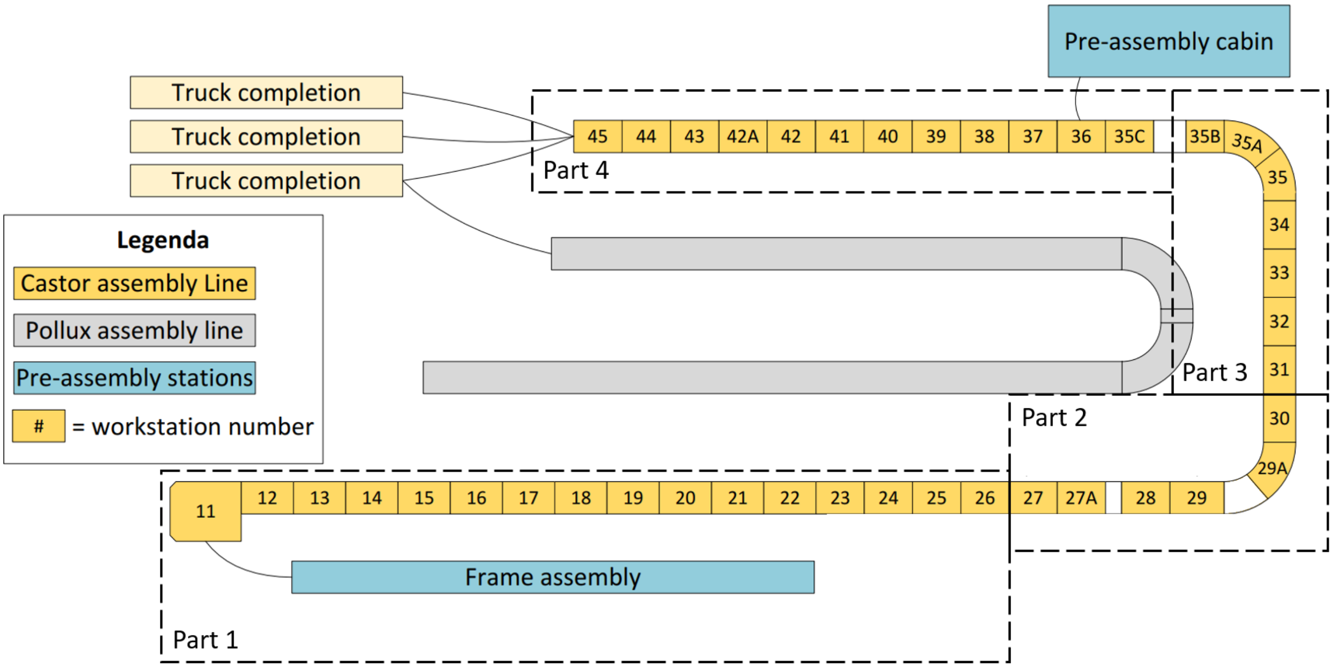


Figure 5.1: Overview of assembly line parts in DES model

We take the following assumptions and simplifications into account:

1. Pre-assembly stations (such as the pre-assembly for the cabin as shown in Figure 5.1) are not explicitly modelled. Line stops caused by these stations are included in the technical stops.
2. The average processing times per task are determined by former analysis at SPZ. The processing time of a truck at a station is based on the maximum processing time of all the random generated processing times of the tasks at that station.
3. Castor1 stops are generated for the entire line, not per station.
4. There are always enough carriers.
5. Processing times at Castor1 always equal the takt time. The technical stoppage time of Castor1 implicitly models processing times that take longer than the takt time.
6. In our model, when a technical stop at Castor1 (Station 26 until 27A) is generated, all stations stop. Then, after the stop, all operators finish their processes. In reality, the operators continue during the stop, and then wait for the station with a failure. However, we do not measure idle time at Castor1, and since we only model Castor1 to recreate realistic arrival rates for Castor2, this difference does not influence our results.
7. When a stop is generated at Castor2 (technical or because of unfinished tasks), all carriers stop, while all stations continue to work on their tasks, and then wait until the carriers continue. Here, we assume that the trucks are always reachable by all equipment on the station.
8. Since we want to focus on the influence of Castor1 and Castor2 on each other, we set the processing times of the *in between-stations*, Station 28 and 29, equal to the takt times for these stations. This means that these stations do not cause stoppages because of long task times and we can focus on the impact of the two line parts on each other. This is a realistic assumption, since these stations are no bottleneck stations.
9. A failure at the end of the day is always fixed at the start of the next day.
10. Breaks are not included in the simulation, because the assembly line is only paused then, and the operators continue with their tasks after the breaks.
11. If the tasks of a long truck are not finished yet when the front of the truck passes the boundary of the next station, the line is only stopped if the rear of the truck is within the station itself. The carrier line should not be stopped while the truck is not entirely within the station yet.

5.1.2 Conceptual Model

This section discusses how we modelled the most complex processes in our DES.

One of these complex processes is stoppages, which can be caused by technical problems, or by tasks that are unfinished while the truck already leaves the station. To model these stoppages, we include a sensor at each left and right boundary of the stations. To illustrate this, we use Figure 5.2.

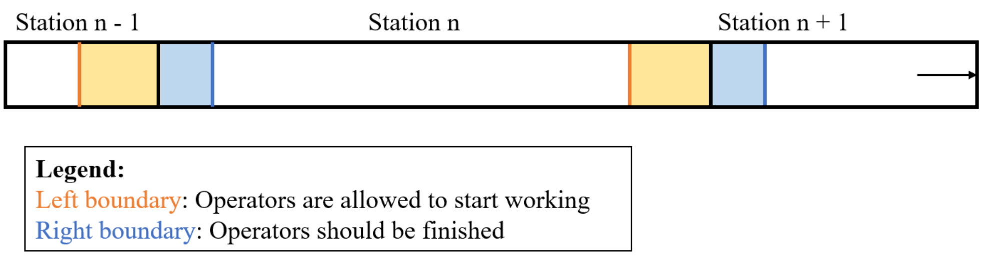
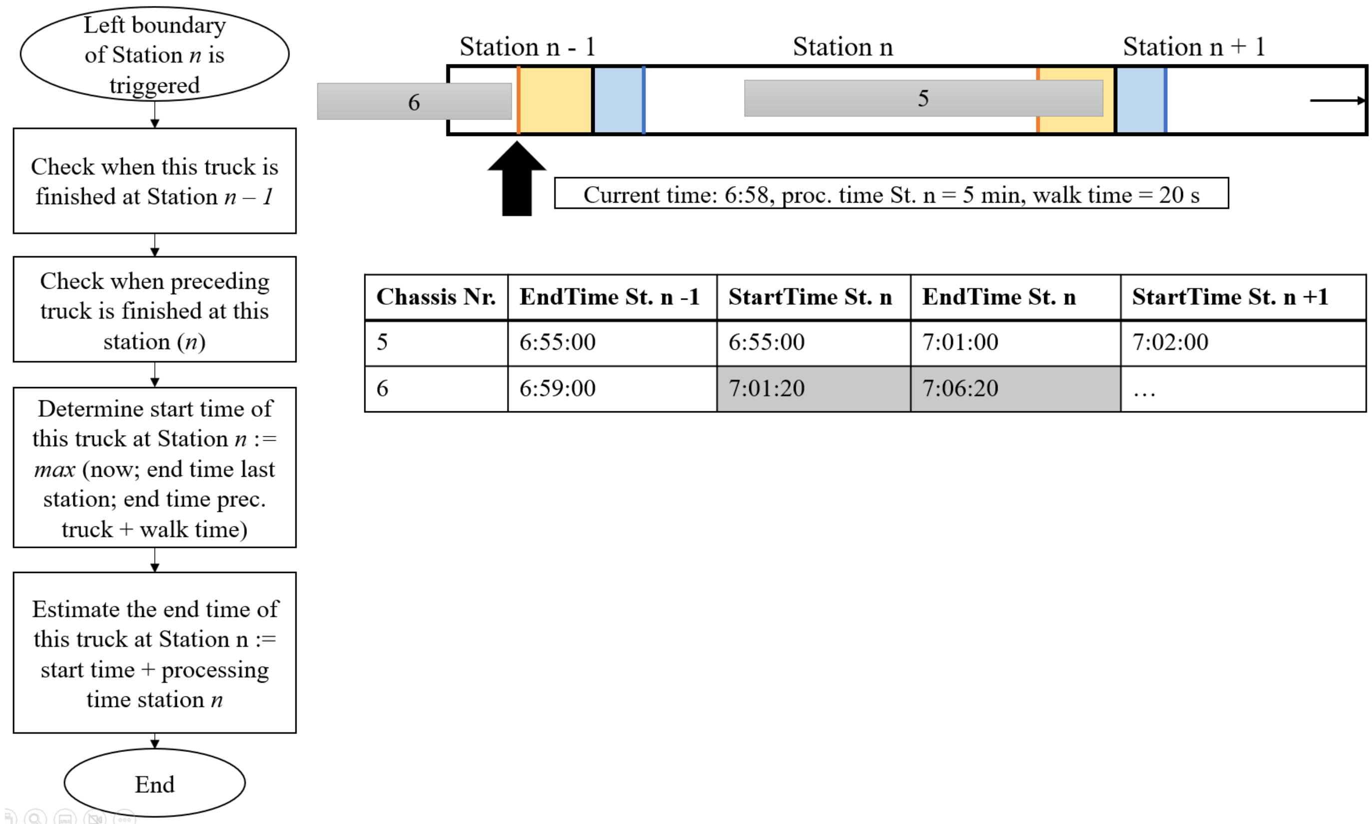


Figure 5.2: Illustration of left and right boundaries of stations

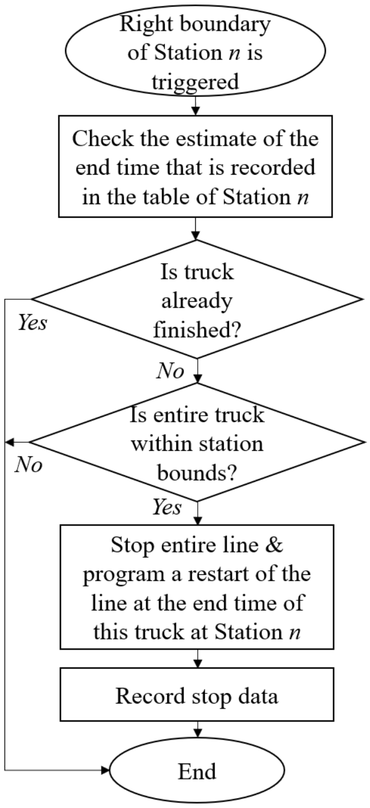
For these stoppages, we record all start and end times of the tasks per truck per station. If the front of the truck passes the left boundary, then we determine when the operators can start with the tasks of the truck. Figure 5.3 shows this process, where the start time of truck 6 is determined. Truck 6 can start if it is finished at station $n - 1$, and if the tasks of truck 5 are finished at station n. Since the operators do not stop with their tasks anymore if they already started, we can already determine the end time of the tasks at station n for truck 6 as well. Figure 5.3 shows in a flow chart how our DES determines the start and end times for all trucks and all stations.

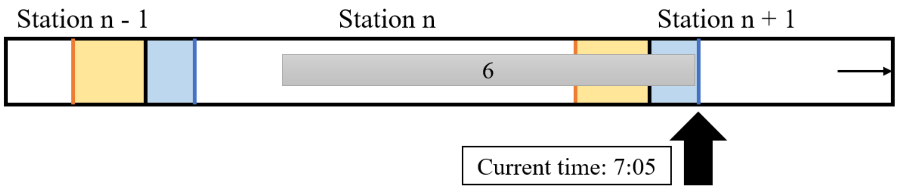


Chassis Nr.	EndTime St. n -1	StartTime St. n	EndTime St. n	StartTime St. n +1
5	6:55:00	6:55:00	7:01:00	7:02:00
6	6:59:00	7:01:20	7:06:20	…

Figure 5.3: Flow chart of left boundary sensor and example

To determine whether the carrier line should be stopped because tasks are not finished yet while a truck already leaves the station, we also include a right boundary sensor. If the front of the truck passes this sensor (see Figure 5.4), we check if the tasks are already finished. If this is not the case, the carrier line should be stopped until the tasks are finished, unless the truck is very long. In this case, it does not make sense to stop the carrier line while the truck is not entirely within the station yet. If the back of the truck enters the left station bound, then this process is called again. We record each stoppage with begin and end times, which Figure 5.4 shows in a flow chart.





Chassis Nr.	EndTime St. n -1	StartTime St. n	EndTime St. n	StartTime St. n +1
5	6:55	6:55	7:01	7:02
6	6:57	7:01	7:06	

Chassis Nr.	Station	Start of Stop	End of Stop
6	Station *n*	7:05	7:06

Figure 5.4: Flow chart of right boundary sensor and example

Another complex process in our model is the placing of a truck on a carrier at Station 28. If a chassis is ready to enter Station 28 and Station 28 is empty, it is placed on a carrier using the flow described in Figure 5.5.

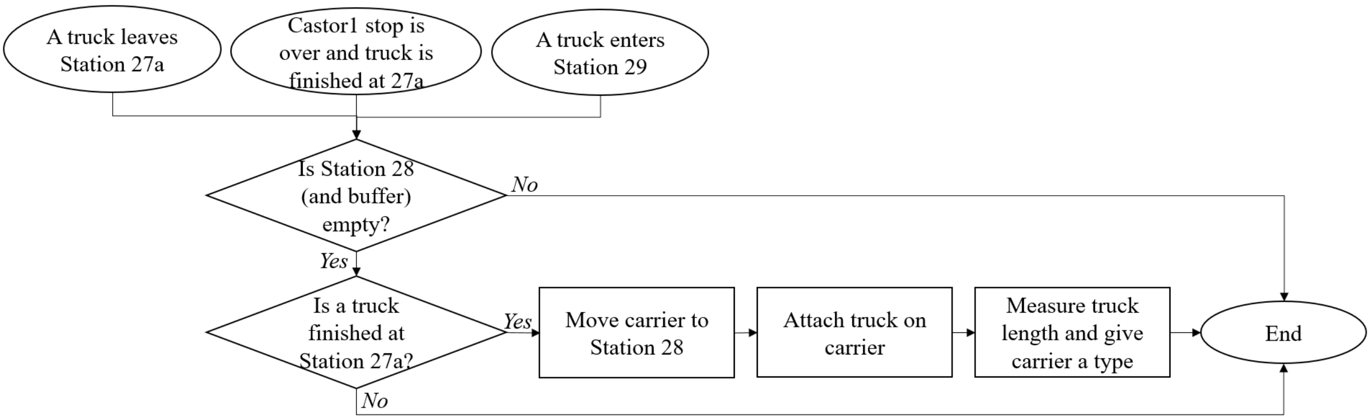


Figure 5.5: Flow chart of placing a truck on a carrier

As explained in Section 2.1, many criteria should be satisfied if a truck is launched on the carrier line. The truck should be released from Station 29, which means that the tasks at Station 28 also should be finished. A truck should also be able to enter the carrier line, which means there should be a safe distance and the line is not stopped. In the current situation, the preceding truck should be at least one takt time at the carrier line, to ensure that a maximum of one truck per takt time is launched on the line. In our *variable rate launching* strategy, this criterion does not apply. In our *triple takt* strategy, this criterion applies for all trucks except those that can compensate a long truck in their group of three trucks. Figure 5.6 describes the flow that the DES follows for the launching.

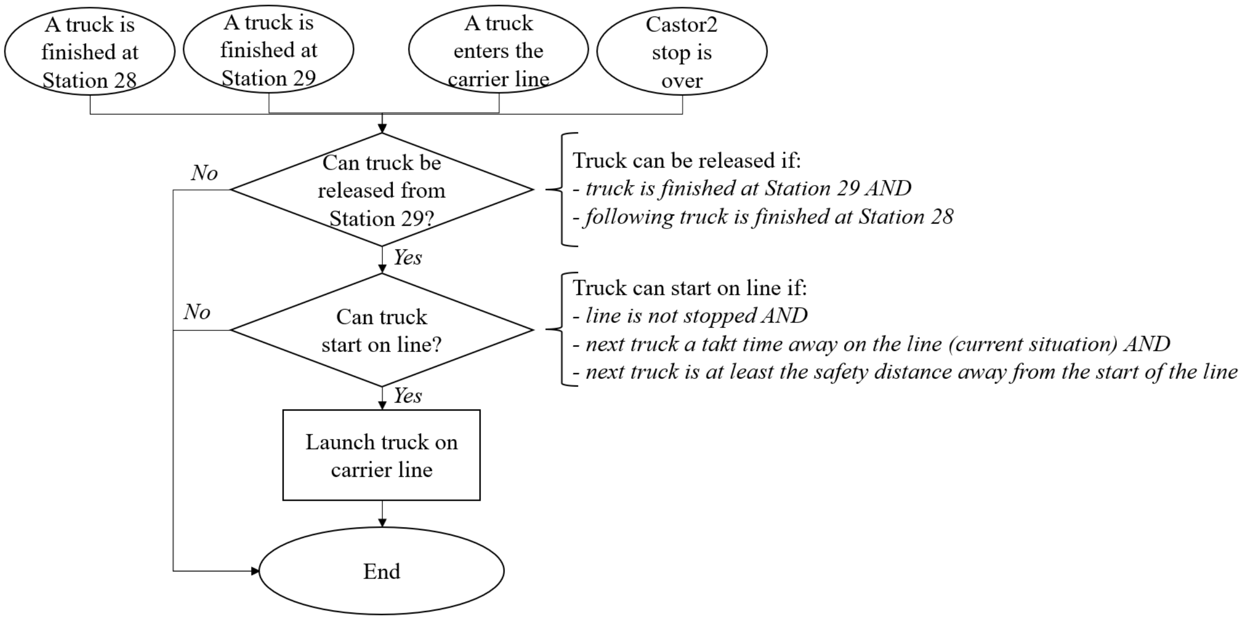


Figure 5.6: Flow chart of launching carrier to carrier line

5.1.3 Model Implementation and Validation

Figure 5.7 shows the actual DES model modelled in the Tecnomatix Plant Simulation Software developed by Siemens Digital Industries Software, version 16.1. In Figure C.1 in Appendix C, the model is run with the settings of the current situation of SPZ, where the blue box shows the outcomes of the main KPIs. This figure also shows the run information: we use a warm-up period of 1 day, and run 5 observations per experiment that exists of 16 days, for which we refer to Appendix A. Figures 5.8, 5.9, 5.10 and 5.11 show all four parts of the simulation model. Figure 5.9 also depicts the trucks on the line, where red, purple and blue represent long, medium and short trucks, respectively.

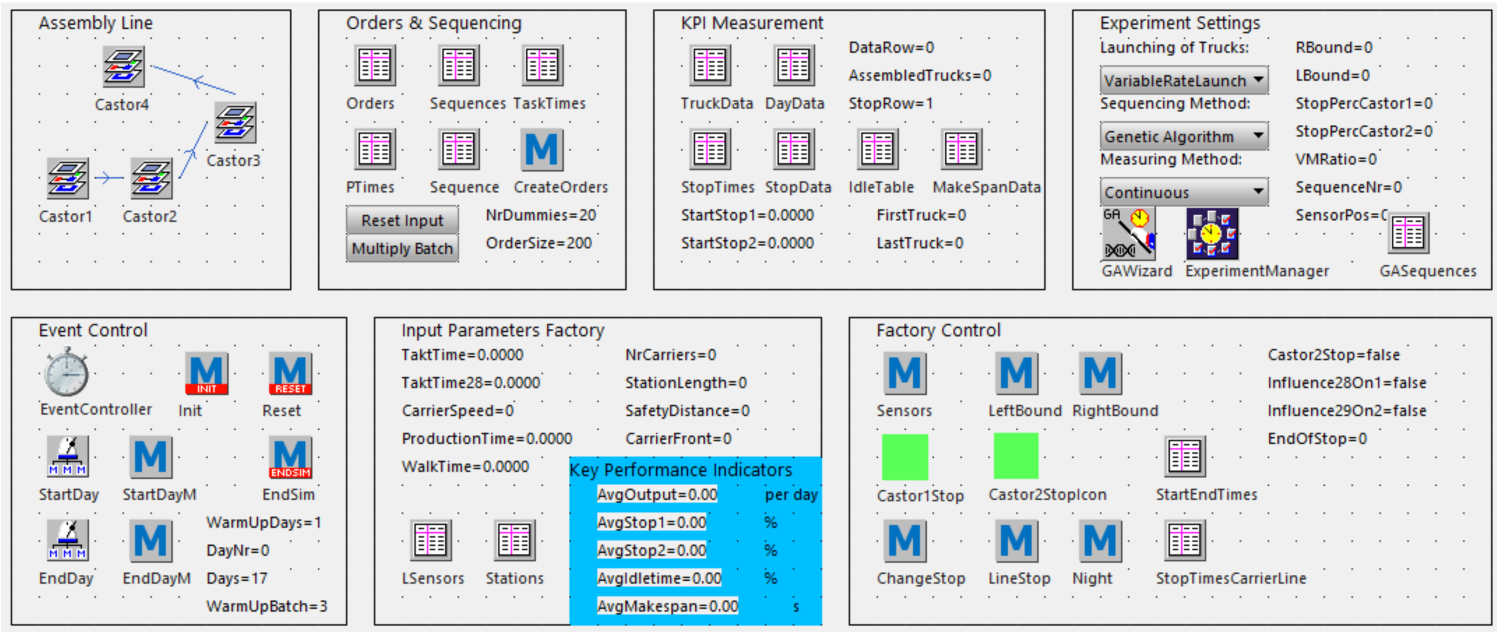


Figure 5.7: DES model implemented in Tecnomatix Plant Simulation software

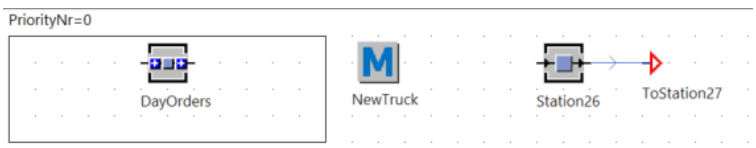


Figure 5.8: First part of Castor DES model

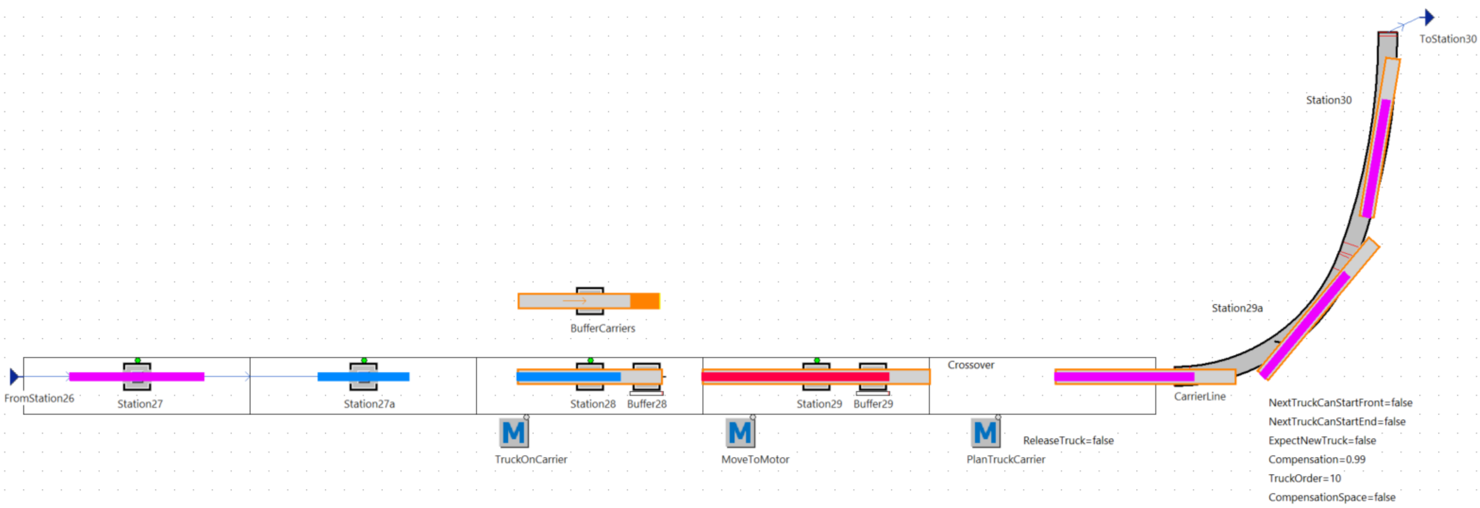


Figure 5.9: Second part of Castor DES model, while running, rectangles represent trucks

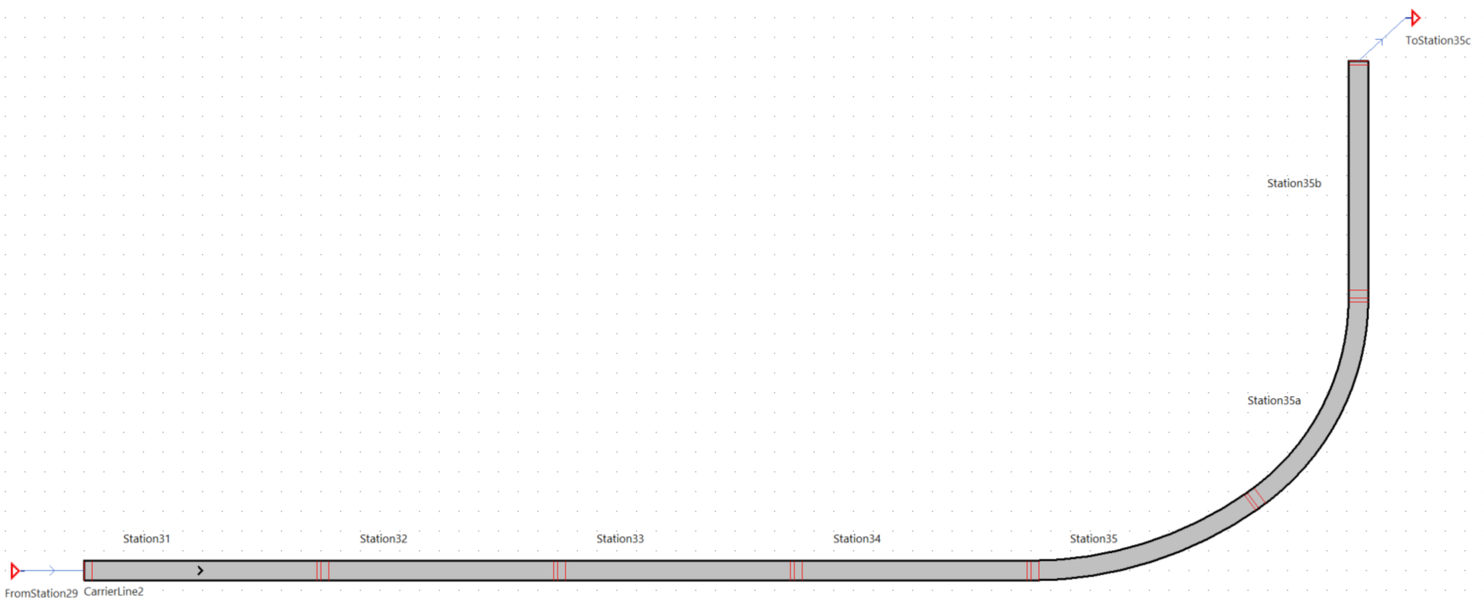


Figure 5.10: Third part of Castor DES model

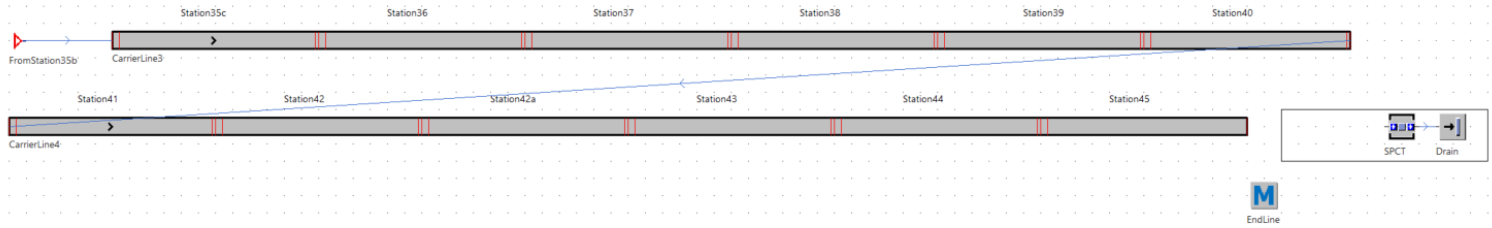


Figure 5.11: Fourth part of Castor DES model

We verified our model using white and black box testing. For white box testing, we discussed the entire model with a simulation expert who is a process engineer at SPZ with knowledge of both Plant Simulation and the assembly line. He confirmed all modelled processes in the simulation, agreed with the assumptions and simplifications and verified that this model represents the actual line sufficiently. For black box testing, we verified the results of our model with the actual production rates using historic data. Table C.4 in Appendix C shows the results from the validation test. The DES model resulted in an average daily output less than the actual output of this order mix. To understand if this is caused by the processing times in the model, we updated the average processing time of all trucks for all stations to the takt time and ran the model again. This resulted in average output closer to the actual data, but with optimistic stoppage times. In consultation with a simulation expert at SPZ we decided that we use this DES model including the processing times, because the model mimics the behaviour of the assembly line sufficiently to draw conclusions regarding the quality of an alternative solution, although it underestimates the absolute production output. For SPZ, we can tune the DES model such that the outcomes do correspond with the actual data. For confidentiality reasons, we use the outcomes that slightly differ from the real production output in this research.

5.2 Experiment Design

This section discusses the design of our experiments to test our alternative solutions compared to the current solution. Section 5.2.1 discusses the generation of the scenarios, and Section 5.2.2 discusses our experiments related to the three genetic algorithm approaches. Section 5.2.3 presents our set of experiments to test the alternative solutions, and Section 5.2.4 presents experiments to perform a sensitivity analysis on these solutions.

5.2.1 Scenario generation

We test the various solutions on real-life data of Scania Production Zwolle. This data consists of processing times of every position of every station, given the configuration of the truck. Research projects at Scania Production Zwolle in 2020 and 2021 collected this data. We assume that this data is still up-to-date, and for the stations that were not included in these projects (29a, 35b and 35c), we randomly generate data based on the takt time, since these stations do not contain bottlenecks recognized by the planning department. On every position, each truck has its own average processing time, based on the variant (for example, axle configuration). We do not have information on the variation of the processing times.

To test the performance of the sequences we generate, we create two data sets based on trucks assembled in June, 2021, one with a few long trucks (*the default order set*), and one with relatively many long trucks(*the long order set*). The data sets include 20 dummy trucks and 200 trucks, since this is more than the daily production rate, but still manageable for our genetic algorithm. The dummy trucks are the preceding trucks before the data set, with a definite sequence. Figure 5.12 shows the truck length distributions of both data sets. To test the performance of the light sensor positions, we use the entire production data of the month June, 2021. In this month, 3080 trucks were assembled on the Castor assembly line, which creates a sufficiently big order set to evaluate the discrete measuring system. For the truck length distribution of the entire month, we refer to Figure 2.5.

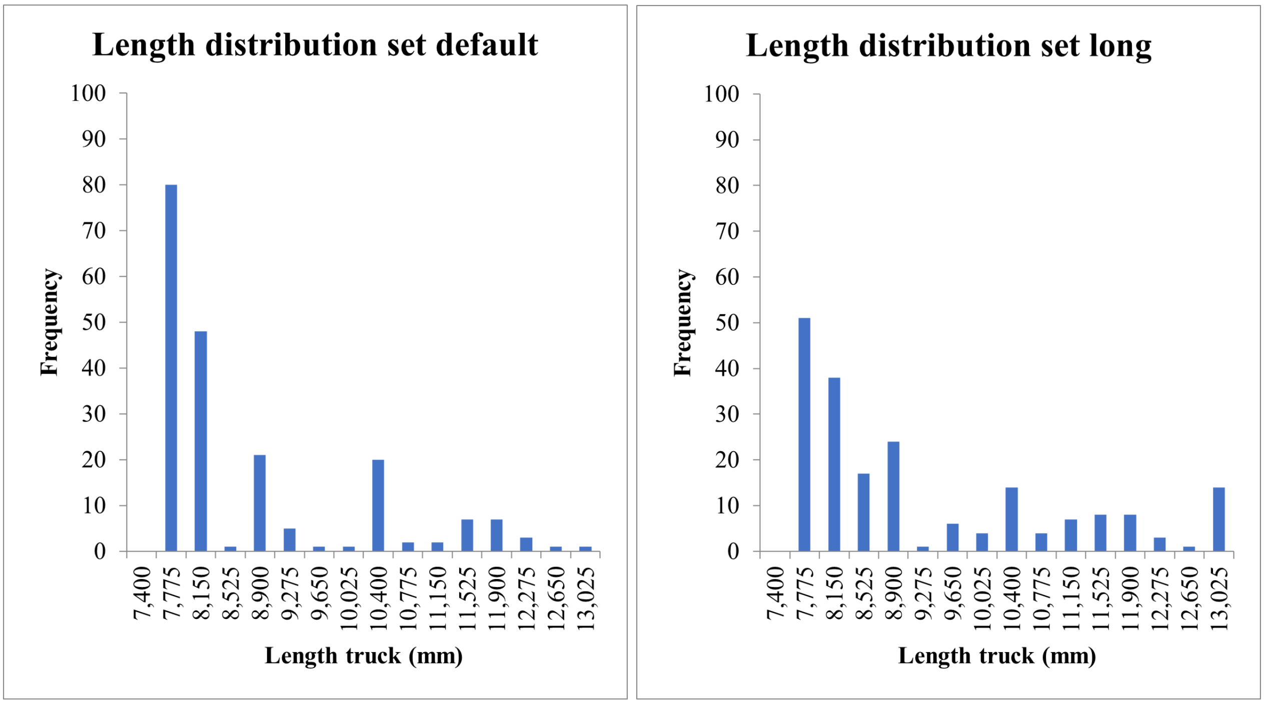


Figure 5.12: Length distributions of order set default vs. long

5.2.2 Genetic Algorithm Parameter Tuning

First, we validate the SE approach using 1000 random sequences that we evaluate with the DES model and the SE approach. We determine the correlation coefficient to conclude whether the SE approach is able to distinguish good sequences. In this way, we can compare the SE approach with the DES and SH approaches and comment on their performance.

Then, we tune the parameters of the GA of all approaches using the experiments in Table 5.1. Since each experiment takes around four hours (in the DES model), we test the combination of crossover and mutation operators and selection of the parents first. After that, we only experiment with the probabilities of the operators for the best combination. We also experiment with *delta probabilities*, which reduces the crossover and/or mutation probability each generation.

Table 5.1 Experiments for GA Tuning

Exp.	Population Size	Number generation	Mutation Prob.	Crossover Prob.	Crossover	Mutation	Fitness Reference
1	50	10	0.2	0.8	PMX	Random	Absolute
2	50	10	0.2	0.8	PMX	Inversion	Absolute
3	50	10	0.2	0.8	Order	Random	Absolute
4	50	10	0.2	0.8	Order	Inversion	Absolute
5	50	10	0.2	0.8	PMX	Random	Relative
6	50	10	0.2	0.8	PMX	Inversion	Relative
7	50	10	0.2	0.8	Order	Random	Relative
8	50	10	0.2	0.8	Order	Inversion	Relative
9	50	10	0.4	0.8	TBD	TBD	TBD
10	50	10	0.1	0.8	TBD	TBD	TBD
11	50	10	0.2	0.6	TBD	TBD	TBD
12	50	10	0.2	1	TBD	TBD	TBD
13	50	10	0.2 (delta 0.05)	0.8 (delta 0.05)	TBD	TBD	TBD
14	50	10	0.3 (delta 0.1)	0.9 (delta 0.1)	TBD	TBD	TBD
15	100	5	0.2	0.8	TBD	TBD	TBD

Finally, we execute the GA of each approach, in order to generate sequences that result in a high production output. For the SE and DES approach, we use the five best performing sequences for the experiments. For the SH approach, we only use the best performing sequence for the experiments, since this approach already evaluates the 100 best sequences in detail.

5.2.3 Experiments

This section discusses our exact and stochastic experiments to evaluate all our alternative solutions and to execute a sensitivity analysis. Table 5.2 shows all our alternative solutions. We compare all these solutions with our benchmark, the current situation. The current situation is based on Lean launching (a truck can only be launched at least one takt time after its preceding one), a sequence determined by a car-sequencing GA and the position of the five current light sensors. For the sequencing method, we test the five best sequences resulted from the GA based on the DES or SE approach, depending on which approach performs best, and the best sequence resulting from the SH approach. For the measuring method, we test both the positions of the current five light sensors, and the optimally placed positions of five up to ten light sensors. We benchmark these solutions with a continuous measuring method.

Table 5.2 Alternative solutions

Solution Description	Alternative Solutions	Number of Solutions
Launching of Trucks	[Lean, Variable rate launching, Triple takt]	3
Sequencing Method	[Current, GA generated]	1 + 5 best seq. + 1
Measuring Method	[Current, Optimally placed light sensors, Continuous]	1 + 6 options + 1

For the optimal number and positions of the light sensors, we execute exact experiments using our LP and stochastic experiments using our DES model. We also use our DES model to evaluate the performance of our GA generated sequences and to analyse the characteristics of well-performing sequences. In the DES model, we include the evaluation of the performance of the sequencing methods as well. This section discusses these experiments in more detail.

Exact Experiments Light Sensors

We perform exact experiments to advise SPZ in the positions and number of the light sensors. For each year of the NTG period, for 5 up to 10 sensors, we use our LP to determine the optimal positions of the light sensors and the following output loss in trucks per year. Since 2019 and 2022 are not fully included in the data set, we also calculate the lost trucks as a percentage of the total yearly production. To understand whether light sensors should be replaced after a new product introduction (that would lead to a different truck length distribution), we also determine the output loss for the years 2020 to 2022, based on the light sensor positions that were optimal given the data from 2019.

Stochastic Experiments Light Sensors & Launching Strategies

Table 5.3 shows the experiments that we execute in the DES model using the data from the entire month June, 2021, to compare the various number and positions of the light sensors, for all three launching strategies. For the variable rate launching strategy, the length of all trucks should be measured, contrary to the current situation where only long trucks were measured. Therefore, we cannot experiment with the current position of the light sensors, and only experiment with the optimal positions for the number of light sensors based on the data of 2021 resulting from our discrete measuring location model (LP).

Table 5.3 Experiments light sensor options

Exp. Nr.	**Launching of Trucks**	**Number of Sensors**
0	Lean	Current
1	Lean	5
2	Lean	6
3	Lean	7
4	Lean	8
5	Lean	9
6	Lean	10
7	Lean	Continuous
8	Triple Takt	Current
9	Triple Takt	5
10	Triple Takt	6
11	Triple Takt	7
12	Triple Takt	8
13	Triple Takt	9
14	Triple Takt	10
15	Triple Takt	Continuous
16	Variable Rate Launching	5
17	Variable Rate Launching	6
18	Variable Rate Launching	7
19	Variable Rate Launching	8
20	Variable Rate Launching	9
21	Variable Rate Launching	10
22	Variable Rate Launching	Continuous

Stochastic Experiments Sequences & Launching Strategies

Table 5.4 shows the experiments that we execute in the DES model to evaluate the current sequence and the GA generated sequences. In this table, the current situation that serves as a benchmark is shown in italics for both order sets. The last column gives the size of each experiment set, which is based on the fact that the GA sequencing method includes the total 6 best sequences generated by the GAs.

Table 5.4 Experiments to test alternative solutions in DES model

Exp. Set Nr.	**Launching of Trucks**	**Sequencing Method**	**Order Set**	**Size set**
0	*Lean*	*Current*	*Default*	*1*
1	Lean	GA generated	Default	6
2	Variable Rate Launching	Current	Default	1
3	Variable Rate Launching	GA generated	Default	6
4	Triple Takt	Current	Default	1
5	Triple Takt	GA generated	Default	6
6	*Lean*	*Current*	*Long*	*1*
7	Lean	GA generated	Long	6
8	Variable Rate Launching	Current	Long	1
9	Variable Rate Launching	GA generated	Long	6
10	Triple Takt	Current	Long	1
11	Triple Takt	GA generated	Long	6
			Total Experiments:	42

Stochastic Experiments Sequence Performance

Next to the performance of the GA generated sequences, we are also interested in the behaviour of various characteristics of sequences. Therefore, we generate 9 sequences with specifics characteristics:

1. Trucks ordered on their length, descending.
2. Trucks ordered on their length, ascending.
3. Long trucks evenly distributed.
4. Trucks ordered on their cumulative processing time over the 5 bottleneck stations, descending.
5. Trucks ordered on their cumulative processing time over the 5 bottleneck stations, ascending.
6. Trucks with a high cumulative processing time over the 5 bottleneck stations (> 5 takt times) evenly distributed.
7. Long and complex trucks evenly distributed, each long truck is followed with a complex truck.
8. Long and complex trucks evenly distributed, each complex truck is followed with a long truck.
9. Long and complex trucks are evenly distributed, but never consecutive.

We evaluate these sequences for the current situation, the triple takt situation with the current light sensors and the variable rate launching situation with continuous measuring. For this evaluation, we use the long order set of trucks, such that the impact of long trucks is more visible.

5.2.4 Sensitivity Analysis

We perform a one-factor-at-a-time sensitivity analysis to analyse the impact of the order set, the bounds of the stations, the stop percentages and variance of the processing times. Table 5.5 shows all experiments that we execute to evaluate the robustness of the solutions, where the blue cells indicate the factor that we change in that experiment. We perform the sensitivity analysis on the following five scenarios:

1. the current situation,
2. the current situation, with a triple takt launching strategy
3. the best performing sequence and position of light sensor, for the triple takt launching strategy,
4. the best sequence resulted by the GA, continuously measuring and the variable rate launching strategy,
5. and the current sequence, continuously measuring and the variable rate launching strategy.

Table 5.5 Sensitivity analysis for results

Experiment Nr.	**Rbound, Lbound work stations**	**Stop Perc. [Castor1, Castor2]**	**VMRatio Proc. Times**	**NrStations**	**Order Set**
0	*[0.5, 0.2]*	*[0.084, 0.084]*	*0.2*	*21*	*Default*
1	[0, 0]	[0.084, 0.084]	0.2	21	Default
2	[1, 0.5]	[0.084, 0.084]	0.2	21	Default
3	[1.5, 1]	[0.084, 0.084]	0.2	21	Default
4	[0.5, 0.2]	[0, 0]	0.2	21	Default
5	[0.5, 0.2]	[0.05, 0.05]	0.2	21	Default
6	[0.5, 0.2]	[0.15, 0.15]	0.2	21	Default
7	[0.5, 0.2]	[0.084, 0.084]	0	21	Default
8	[0.5, 0.2]	[0.084, 0.084]	0.1	21	Default
9	[0.5, 0.2]	[0.084, 0.084]	0.3	21	Default
10	[0.5, 0.2]	[0.084, 0.084]	0.2	2	Default
11	[0.5, 0.2]	[0.084, 0.084]	0.2	9	Default
12	*[0.5, 0.2]*	*[0.084, 0.084]*	*0.2*	*21*	*Long*
13	[0, 0]	[0.084, 0.084]	0.2	21	Long
14	[1, 0.5]	[0.084, 0.084]	0.2	21	Long
15	[1.5, 1]	[0.084, 0.084]	0.2	21	Long
16	[0.5, 0.2]	[0, 0]	0.2	21	Long
17	[0.5, 0.2]	[0.05, 0.05]	0.2	21	Long
18	[0.5, 0.2]	[0.15, 0.15]	0.2	21	Long
19	[0.5, 0.2]	[0.084, 0.084]	0	21	Long
20	[0.5, 0.2]	[0.084, 0.084]	0.1	21	Long
21	[0.5, 0.2]	[0.084, 0.084]	0.3	21	Long
22	[0.5, 0.2]	[0.084, 0.084]	0.2	2	Long
23	[0.5, 0.2]	[0.084, 0.084]	0.2	9	Long

5.3 Conclusion

To conclude, we develop a DES model to evaluate alternative solutions. We design experiments to test all alternative solutions and compare these with the current situation to evaluate their performance. For these alternative solutions, we also design a sensitivity analysis to evaluate the robustness of the alternative solutions.

Chapter 6

Results

This chapter presents the results of all experiments presented in Section 5.2. Section 6.1 presents the comparison of the SE approach and the DES approach presented in this research. Section 6.2 shows the performance of our alternative solutions.

6.1 Comparison SE vs. DES Approach

We experiment with three GA approaches, an approach that mainly focused on optimisation (SE), an approach that mainly focused on evaluation (DES) and an approach that balanced these two (SH). In this section, we validate the simplified evaluation method of the SE approach, with the already validated and verified DES model, that is used in the DES and SH approach.

We generate 1000 random sequences, which we evaluate with both models, for which the outcomes are shown in Figure 6.1. This shows that there is no clear correlation between the outcomes from the two models and the outcomes from the SE model do not correspond with DES model results. This section explains in detail why the SE model is not able to correctly determine the makespan of an order set.

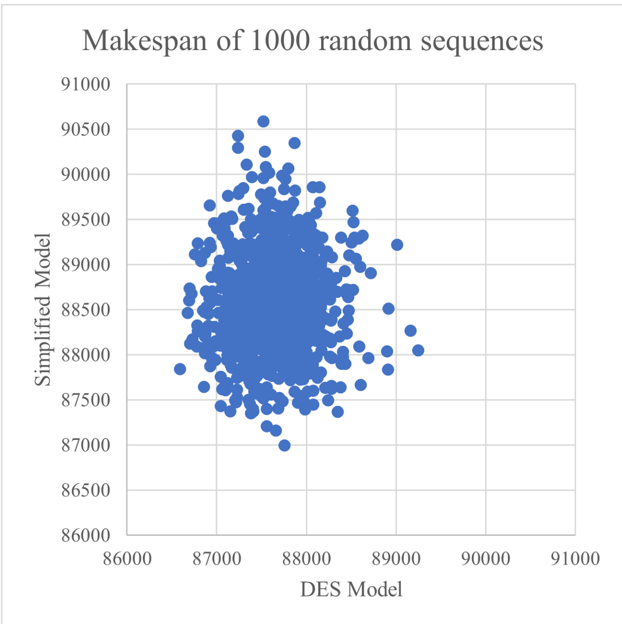


Figure 6.1: Correlation between outcomes of both simulation models

The DES model uses a sequence of events in time to model the assembly line. The simulation time jumps to the next event, and pauses the simulation time at each event, to enable events to happen simultaneously. Our created SE model evaluates the makespan of an order set iteratively, starting from the first truck at the first station. So, if truck 2 causes a stop that influences truck 1, we have to update the already estimated leave time of truck 1. For each truck, we determine the start and end time of the operations per station, and we record the stop times, which are shown in Figures 6.2 and 6.3 for the SE and DES model, respectively. The models differ in two ways, as explained below.

First, the DES model calls events with the same time mark in the order in which the events were created. In our DES example (see cells marked orange), truck 1 enters Station 32 at 634.9 seconds, after which truck 2 leaves Station 30 at 634.9, which creates a stop until 969.8, because truck 2 is not finished yet. In our SE example, truck 1 can only start at Station 32 after the stop, so at 969.8. At the next station (32), both models are at the same time mark again (1269.8), because the DES model still includes the stop time (cells marked green). However, this difference between the models does sometimes cause different start or end times, resulting in different makespans.

Second, the SE model is not able to include stops in a correct way, which we show using the difference of truck 1 starting at Station 34, which is 1608.8 in the SE model, and 1614.6 in the DES model (see cells marked blue). Truck 1 enters Station 33 at 1269.8, and would leave it again at 1569.8, excluding any stops. However, the line is both stopped from 1269.8 to 1304.7 and from 1569.8 to 1579.7. This means that truck 1 leaves Station 33 at 1269.8 + 300 + (1304.7 - 1269.8) + (1579.7 - 1569.8) = 1614.6. Since the processing time of truck 1 equals 339 seconds, the operations are finished at Station 32 at 1614.6 and the operators of Station 33 can start at truck 1. The SE model determines this in the following steps:

1. Truck 3 crosses the end of Station 30 at 1269.8, but is only finished at 1304.7, so the model records a stop of the line.
2. Truck 1 can start at 1269.8 at Station 32, which means it ends at 1608.8 (processing time equals 339 seconds). Including the already recorded stop of truck 3 at Station 30, the truck would leave the station at 1604.7. This is too early to finish all the operations, so the model records another stop until 1608.8.
3. Truck 4 leaves Station 30 at 1569.8, but the operators are only finished at 1579.7, so the models records another stop.

The stop caused by truck 4 at Station 30 is not included anymore in the end time of truck 1 at Station 33, although it does influence it. If the SE model would include this, truck 1 does not need to stop anymore at Station 33, because it is already finished due to the stoppage times of the other trucks. However, this stoppage time might be already used to determine start and end times for other trucks. This example shows that the SE model is not able to include stops in their chronological order, but only in the order they are determined, and cannot consider stoppages simultaneously, but only one stoppage at a time.

Start & End Times												
Chas. \ St.	29		30		31		32		33		34	
1	0	290	300	634.9	634.9	934.9	969.8	1269.8	1269.8	1608.8	1608.8	1985.8
2	300	590	634.9	969.8	969.8	1244.8	1269.8	1569.8	1608.8	1974.8		
3	600	890	969.8	1304.7	1304.7	1604.7	1608.8	1908.8				
4	900	1190	1304.7	1579.7	1604.7	1879.7						
5	1200	1490	1579.7	1914.6								
6	1500	1790										
7												
8												

Stop list			
Truck	Station	Start	End
1	30	600	634.9
2	30	934.9	969.8
3	30	1269.8	1304.7
1	33	1604.7	1608.8
4	30	1569.8	1579.7
1	34	1908.8	1985.8
2	33	1908.8	1974.8
5	30	1844.8	1914.6

Figure 6.2: Start, end and stop times of SE model

Start & End Times												
Chas. \ St.	29		30		31		32		33		34	
1	0	290	300	634.9	634.9	934.9	934.9	1234.9	1269.8	1608.8	1614.6	1991.6
2	300	590	634.9	969.8	969.8	1244.8	1269.8	1569.8	1614.6	1980.6		
3	600	890	969.8	1304.7	1304.7	1604.7	1614.6	1914.6				
4	900	1190	1304.7	1579.7	1604.7	1879.7						
5	1200	1490	1579.7	1914.6								
6	1500	1790										
7												

Stops (first 20)			
Truck	Station	Start	End
1	30	600	634.9
1	31	934.9	934.9
2	30	934.9	969.8
3	30	1269.8	1304.7
4	30	1569.8	1579.7
5	30	1844.8	1914.6
1	34	1984.4	1991.6

Figure 6.3: Start, end and stop times of DES model

To conclude, these two differences cause an incorrect calculation of the makespan by the SE model, which is not consistently higher or lower than the makespan determined by the DES model. The incorrect calculation is caused by the SE model that only considers stoppages one at a time, which deviates from the reality modelled in the DES model. Therefore, we are not able to use the SE model, since it cannot estimate the quality of a sequence sufficiently.

6.2 Alternative Solutions Performance

This section discusses the performance of our designed alternative solutions, based on the experiments presented in our experiment design. Section 6.2.1 presents the performance of the sequences generated by the DES and the SH GA approach. Section 6.2.2 shows the performance of our alternative solutions based on exact and stochastic experiments, and Section 6.2.3 presents the sensitivity analysis of these solutions. The GA tuning experiments were run on a computer equipped with an Intel 2.60 GHz and 16 GB of RAM. All other experiments are run on a computer equipped with an Intel 1.60 GHz and 8 GB of RAM and the LP experiments are solved with Gurobi.

6.2.1 Genetic Algorithm Results

We tune the parameters of the GA in the DES model for the long order set, after which we compare the alternative solutions for both order sets in the DES model, as described in Section 5.2.2.

Table 6.1 shows the GA tuning experiments from the long order set, which all have a relatively long run time. The differences between the experiments are small, but experiment 8, with a combination of an order crossover and inversion, results in the best output. Figure 6.4 presents the performance graph of this experiment, which shows that the worst sequence in the generation improves rapidly, but the best sequence improves only slowly. An explanation could be that the algorithm struggles with escaping local optima, although increasing the mutation probability (see exp. 9 or 15) to increase the diversification, does not seem to help.

Table 6.1 GA tuning results order set long

Exp	**Population Size**	**Number generation**	**Mut. Prob.**	**CO Prob.**	**CO**	**Mutation**	**Fitness Ref.**	**Time**	**Best Output**
1	50	10	0.2	0.8	PMX	Random	Absolute	2:04:53	138.48
2	50	10	0.2	0.8	PMX	Inversion	Absolute	2:50:59	138.23
3	50	10	0.2	0.8	OX	Random	Absolute	5:31:49	138.38
4	50	10	0.2	0.8	OX	Inversion	Absolute	3:02:02	138.38
5	50	10	0.2	0.8	PMX	Random	Relative	5:00:27	138.38
6	50	10	0.2	0.8	PMX	Inversion	Relative	4:20:11	138.38
7	50	10	0.2	0.8	OX	Random	Relative	3:52:11	138.40
8	50	10	0.2	0.8	OX	Inversion	Relative	5:47:5	138.52
9	50	10	0.4	0.8	OX	Inversion	Relative	3:41:10	138.42
10	50	10	0.1	0.8	OX	Inversion	Relative	4:0:24	138.46
11	50	10	0.1	0.7	OX	Inversion	Relative	3:42:30	138.42
12	50	10	0.2 (delta 0.05)	0.8 (delta 0.05)	OX	Inversion	Relative	3:6:55	138.42
13	50	10	0.2	0.6	OX	Inversion	Relative	3:20:8	138.42
14	50	10	0.2	1	OX	Inversion	Relative	5:31:49	138.38
15	50	10	0.3 (delta 0.1)	0.9 (delta 0.1)	OX	Inversion	Relative	2:44:35	138.23
16	100	5	0.2	0.8	OX	Inversion	Relative	3:31:39	138.33

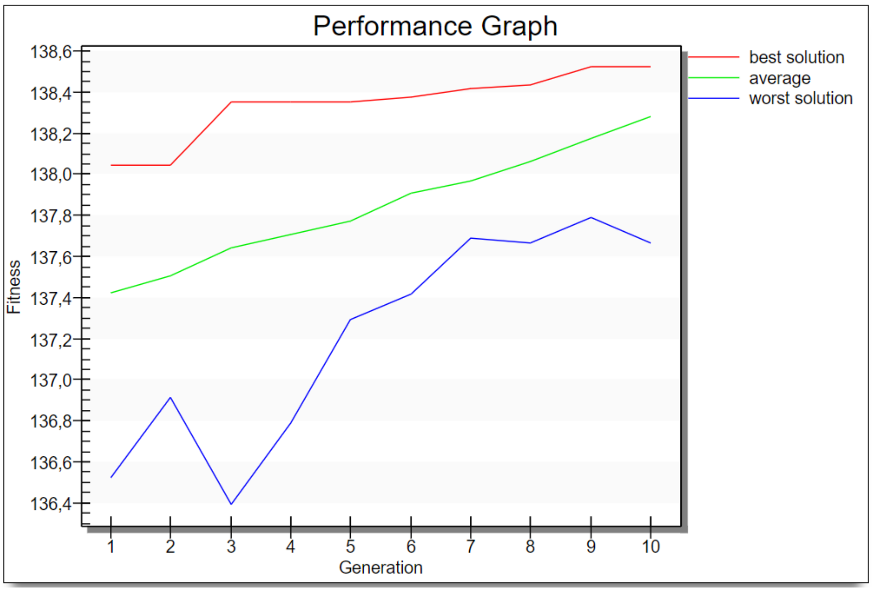
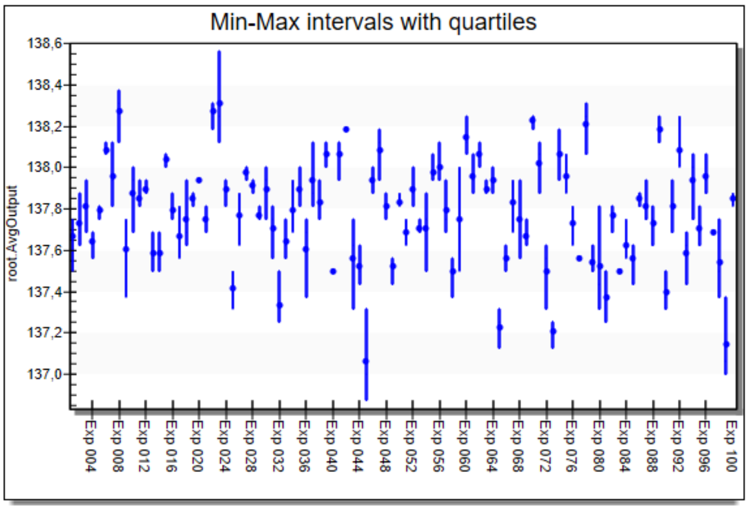


Figure 6.4: Performance GA of exp. 8 of the parameter tuning experiments order set long

Next to the DES approach, we also used our SH approach. We run the deterministic GA with 10 generations and a population size 100, with the parameters of experiment 8 in Table 6.1, with the goal of minimizing a combination of the makespan and the idle time percentage with equal weight. Both these KPIs correlate with our main KPI, the average daily output. The run time of this deterministic GA was around 16 and 9 minutes for the long and default order sets, respectively. Figure 6.5 shows the outcomes of the evaluation of all 100 sequences with their min-max quartiles for the long and default order set, which both took around 20 minutes. This means that in about 30 minutes we are able to find a good sequence using simheuristics with comparable results as the GA in the DES model (see Table 6.1).



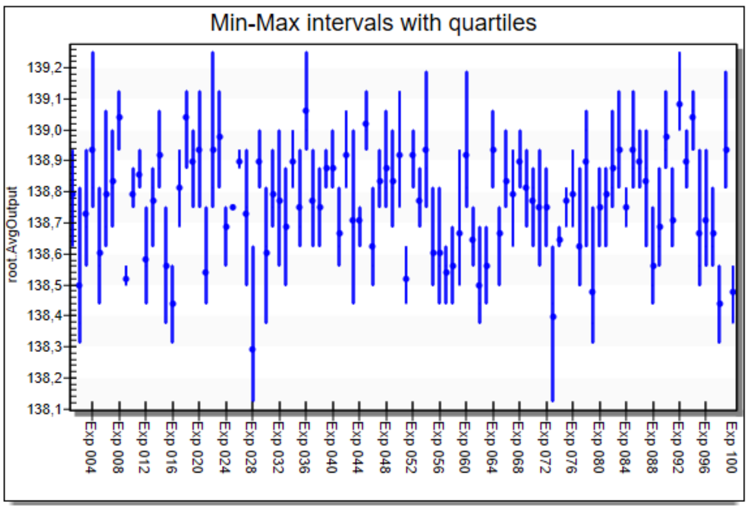


Figure 6.5: Min-max intervals from the best 100 sequences resulting from the deterministic simulation of the long (left) and default (right) order set

Table 6.2 shows the five best sequences resulting from the GA run in Plant Simulation with a population size of 50, 10 generations, an inversion probability of 0.2 and an order crossover, for both order sets. It also includes the best sequence of the SH approach (see Figure 6.5). The fitness (average daily production output) of the GA is based on three observations, but we add an analysis of five observations for the stoppage and idle time percentages, as Appendix A explains. Figure 6.6 shows the min-max intervals with quartiles of the average daily output per sequence for both order sets.

Table 6.2 Best sequences resulting from the GA, DES and SH approach

Seq.	Order Set	Individual	Fitness (3 obs.)	Fitness (5 obs.)	Stop Castor1 (%)	Stop Castor2 (%)	Idle Time (%)
1	Long	Gen 8 Ind 94	138.52	138.49	16.2	38.7	23.5
2	Long	Gen 9 Ind 7	138.52	138.48	16.2	38.7	23.5
3	Long	Gen 10 Ind 81	138.50	138.49	16.2	39.0	23.5
4	Long	Gen 9 Ind 66	138.46	138.45	16.1	38.6	23.5
5	Long	Gen 10 Ind 13	138.46	138.48	16.4	38.1	23.5
6	Long	23rd Seq.	138.31	138.28	16.2	39.4	23.5
1	Default	Gen 10 Ind 61	139.21	139.18	14.8	41.6	22.8
2	Default	Gen 4 Ind 78	139.15	139.11	15.0	42.2	22.8
3	Default	Gen 9 Ind 26	139.15	139.14	14.3	41.0	22.8
4	Default	Gen 9 Ind 79	139.15	139.13	14.2	42.1	22.8
5	Default	Gen 1 Ind 8	139.13	139.10	15.0	42.1	22.8
6	Default	92nd Seq.	139.08	139.09	15.0	41.3	22.9

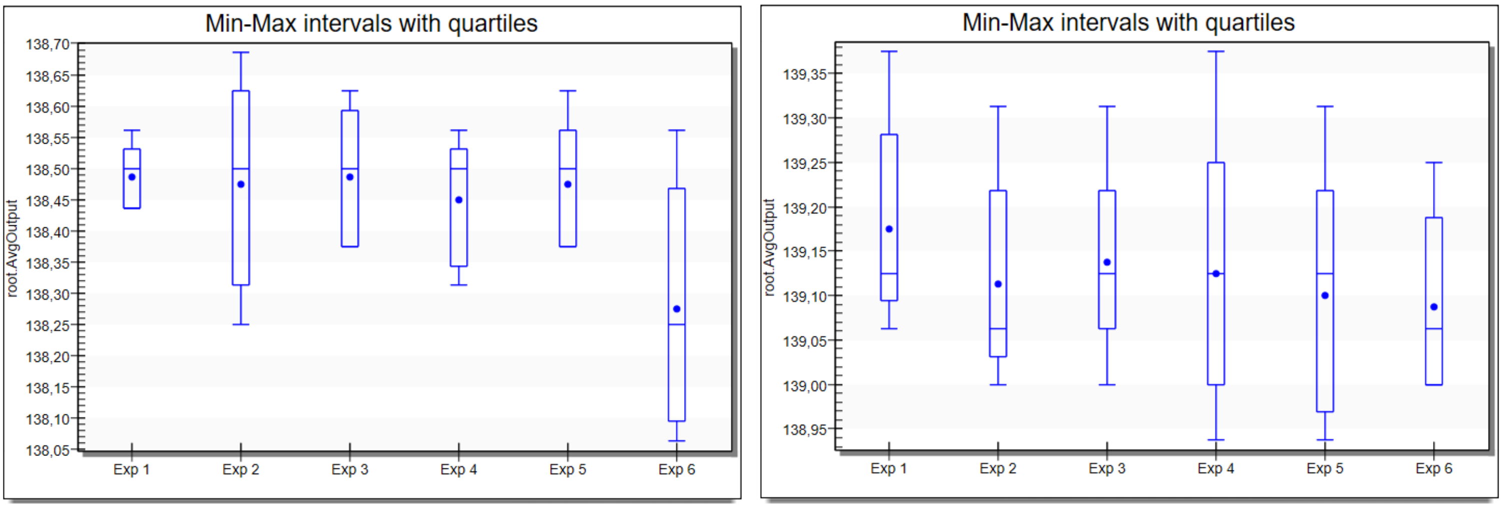


Figure 6.6: Min-max intervals with quartiles of the average daily output per sequence of long (left) and default (right) order set.

6.2.2 Experiment Results

This section presents all outcomes of the experiments described by Section 5.2, in order to advise SPZ on their truck length measuring method, launching strategy and sequencing approach.

Exact Experiments Light Sensors

Figures 6.7 and 6.8 show the output loss for each year per number of light sensors, as a percentage of the total yearly production and in number of trucks per year, respectively. The results of the years in which not all data is available are corrected to full years. Although some years allow for more optimisation than others, these results show that the repositioning of the current (5) light sensors already results in the production of more trucks per year. We also conclude that the improvement possibility caused by increasing the number of light sensors decreases with every added light sensor.

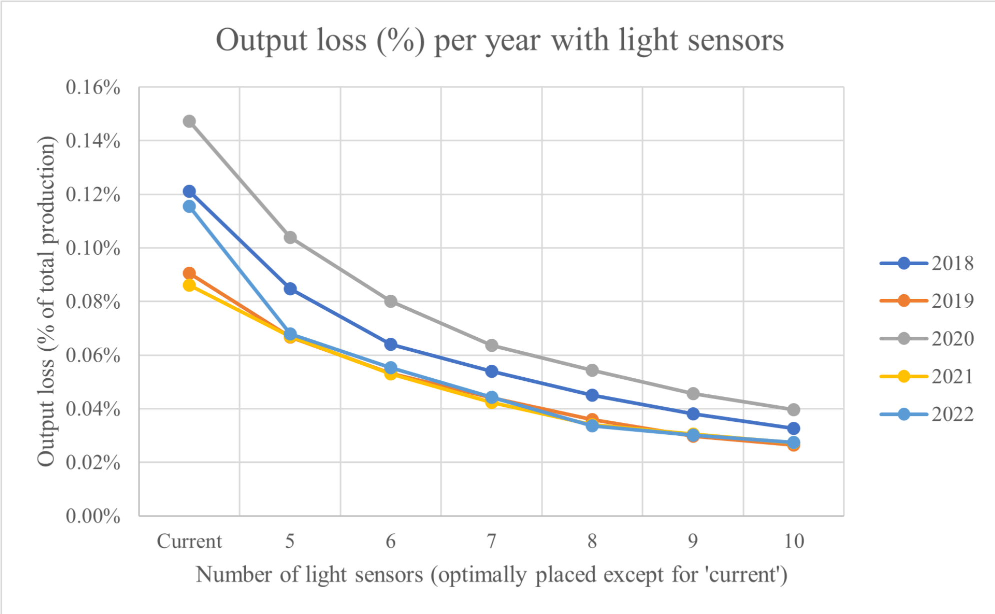


Figure 6.7: Output loss percentage of total production per year, based on the LP determined optimal position per number of light sensors

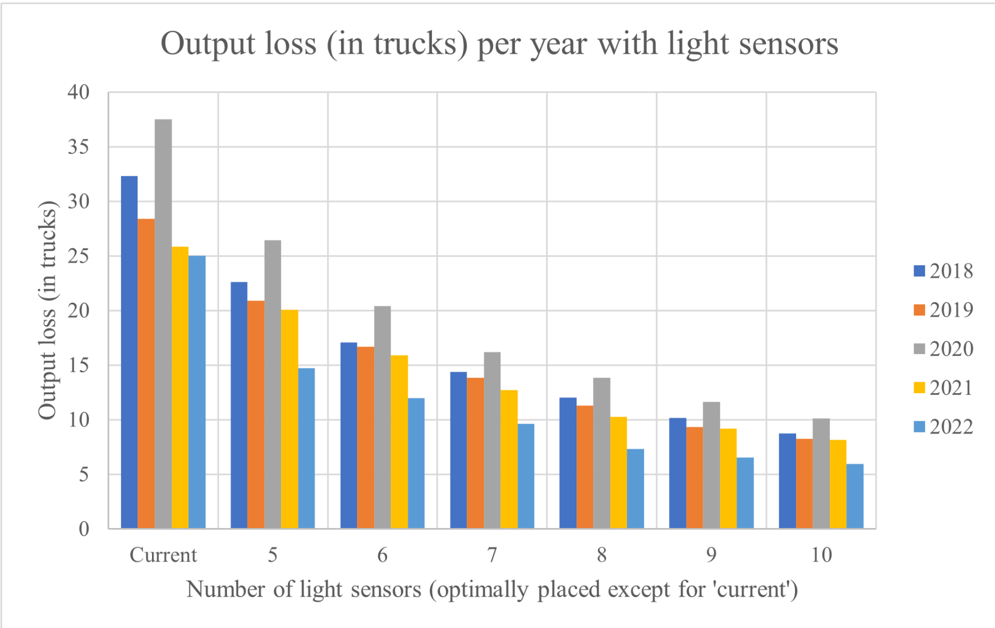


Figure 6.8: Output loss in number of trucks per year, based on the LP determined optimal position per number of light sensors

Since the truck length distribution is not known when the light sensors are placed, SPZ is also interested in the resulting output loss if the light sensors are positioned using historic data. Figures 6.9 and 6.10 show the output loss if the light sensors are placed based on data from 2019 compared to actual data (e.g. based on data from 2020, for the year 2020), as a percentage of the total yearly production and in number of trucks per year, respectively. We test this for both 5 and 10 light sensors. From this experiment, we conclude that the current output loss could even be reduced with positions based on historic data, but positions based on actual data lead to a higher reduction.

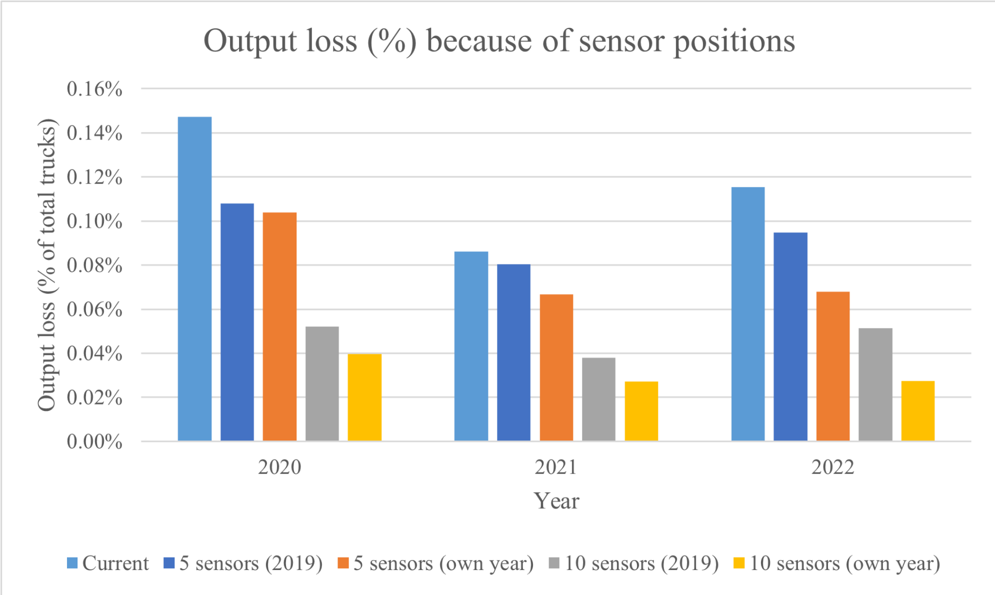


Figure 6.9: Output loss percentage of total production per year, based on the optimal positions of light sensor with 2019 data

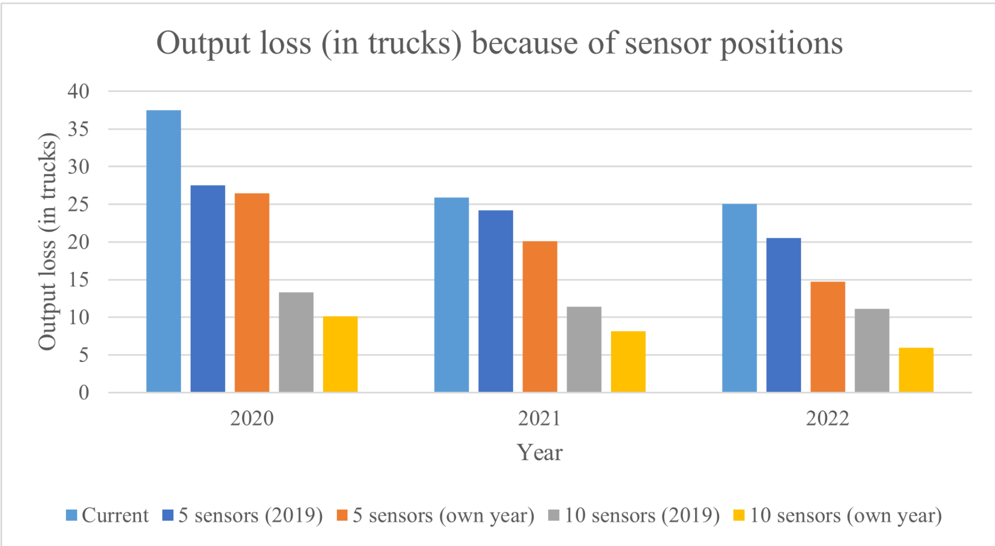


Figure 6.10: Output loss in number of trucks per year, based on the optimal positions of light sensor with 2019 data

Stochastic Experiments Light Sensors & Launching Strategies

We execute the experiments from Table 5.3 to see the best performing light sensor positions for each launching strategy, based on the total production of June, 2021. Figure 6.11 presents the outcomes of these experiments. For the calculations behind the significance of the results, we refer to Appendix B. The results show that, independent of the measuring method, the variable rate launching strategy gives a significant increase in the production output. The triple takt strategy only results in slightly better outcomes and results in the most improvement if the truck lengths are measured continuously. In the current situation (Lean), repositioning the light sensors could increase the monthly production with 3 trucks (0.10%), which is the case with 9 or 10 optimally placed light sensors. The total possible increase is 4 trucks (0.13%), if the truck lengths are measured continuously. However, both these increases are not statistically significant, since the fluctuation in the daily production output is quite high. The exact calculations mentioned before in this section determined that the maximum possible increase in the year 2021 consisted of 18 trucks (see Figure

6.8), which is lower than these stochastic results, which show an increase of 3 trucks per month (on average 36 per year). Figure 6.11 shows that the current situation (Lean launching strategy, five light sensors) could be improved from a monthly production of 2825 trucks to at most 2916 trucks (a 3.2% increase), if the launching strategy is changed to VRL and if the truck lengths are continuously measured. If the lengths are measured discretely, with 10 optimally placed sensors, the VRL strategy realizes a 3.0% increase compared to the current production.

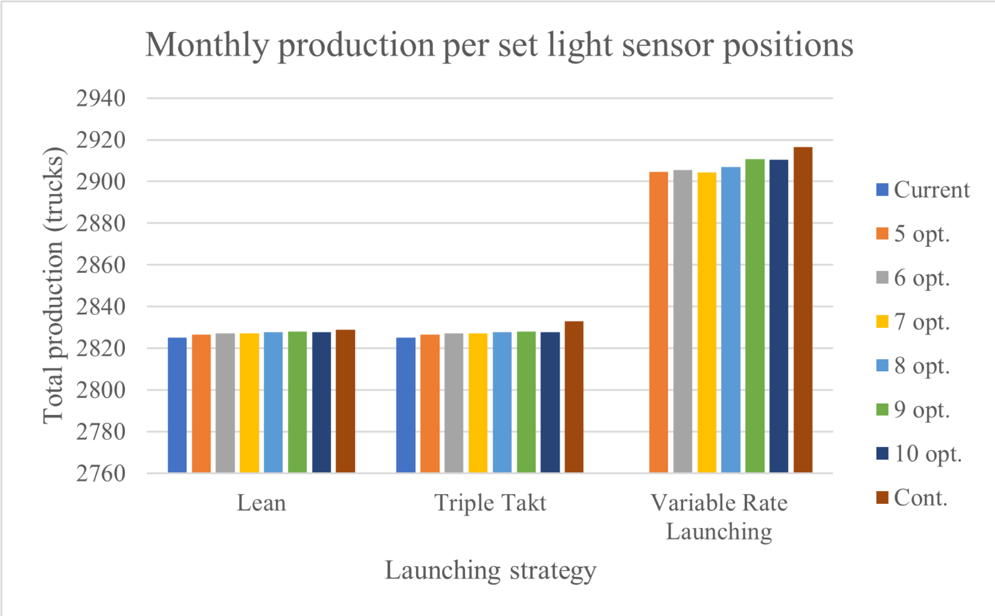


Figure 6.11: Average daily output of the alternative solutions on long order mix experiments

Stochastic Experiments Sequences & Launching Strategies

We also perform experiments to analyse the best performing sequencing method, given all launching strategies, as defined in Table 5.4. In all these experiments, the continuous measuring method is used, to exclude the influence of the light sensor positions. Figure 6.12 presents the outcomes of these experiments. Similar to the light sensor experiments, we are able to conclude that the variable rate launching strategy leads to the highest average daily production, which is a significant increase from the current situation (see Appendix B). The triple takt strategy leads to a small increase of the daily production (0.63% and 0.12% for long and default order set, resp.), but this increase is only significant for the long order set. For each launching strategy, there is no significant difference between the current situation and the best/worst sequence in these experiments. However, an interesting outcome is that the current sequence gives a higher result than the sequences generated by the GA. This could suggest that the performance of the GA is suboptimal. Sequence 6, which is generated by the SH approach, also has no significant difference with the other GA generated sequences, while the generation took less run time than the DES approach. To conclude, these experiments show that the daily production rate of the current situation (Lean, current sequence), could be increased from 134.3 to 138.6 trucks (VRL, current sequence) for the long order set, which is a 3.2% increase. For the default order set, this is an increase of 2.7%.

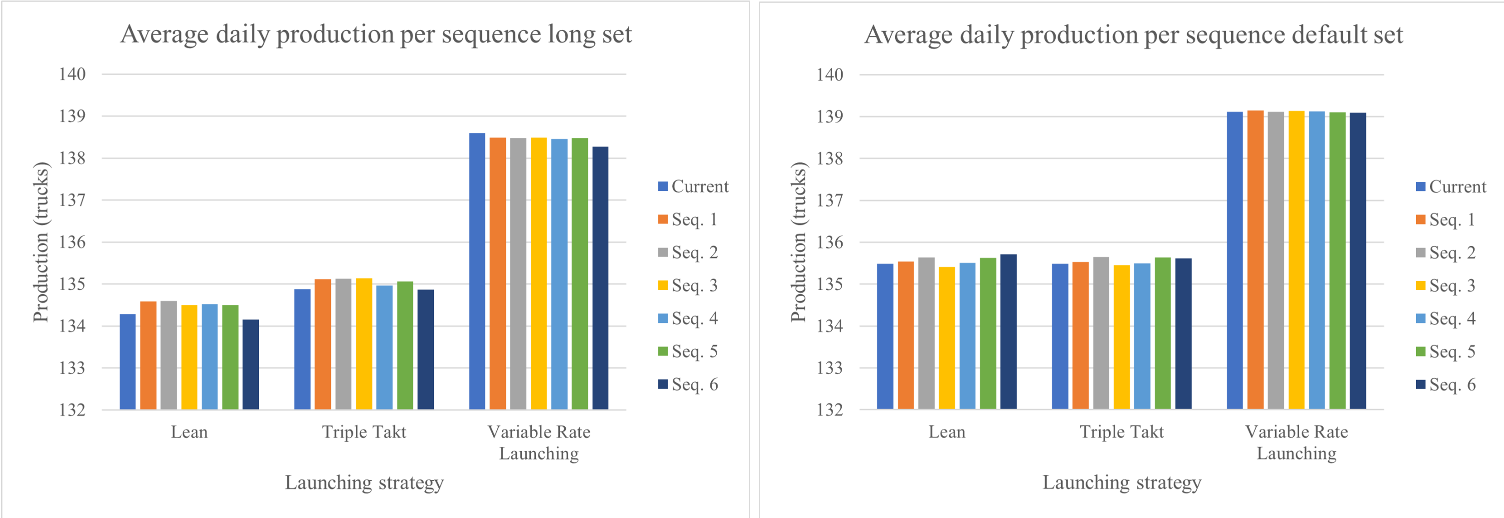


Figure 6.12: Average daily production output of the generated sequences for the launching strategies

Figures 6.13, 6.14 and 6.15 show the stoppage and idle time percentages from the light sensor and sequences experiments, which give similar results. We are not able to make clear conclusions on the difference between the Lean and the triple takt strategy influence on the stoppage and idle time. The impact of the light sensor positions or sequences on the stoppage and idle time is also insignificant. We conclude that the variable rate launching strategy increases the stoppage time at Castor2 significantly. However, the stoppage time at Castor1 decreases, which suggests that the stops at Castor2 are short enough to have no impact on Castor1. The idle time also decreases slightly for this strategy, which means that the work experience for the operators at Castor2 improves.

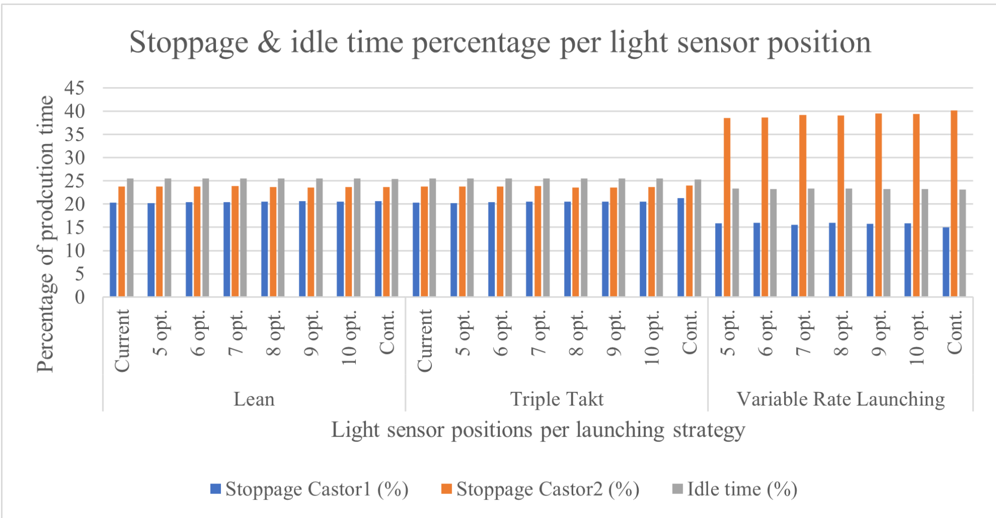


Figure 6.13: Average stop and idle times of the alternative solutions on measuring method experiments

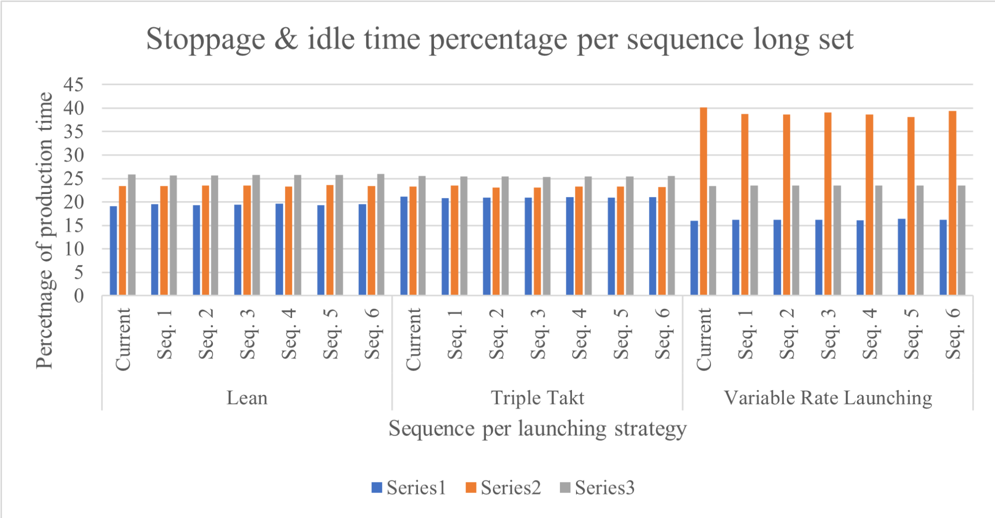


Figure 6.14: Average stop and idle times of the alternative solutions on long order mix experiments

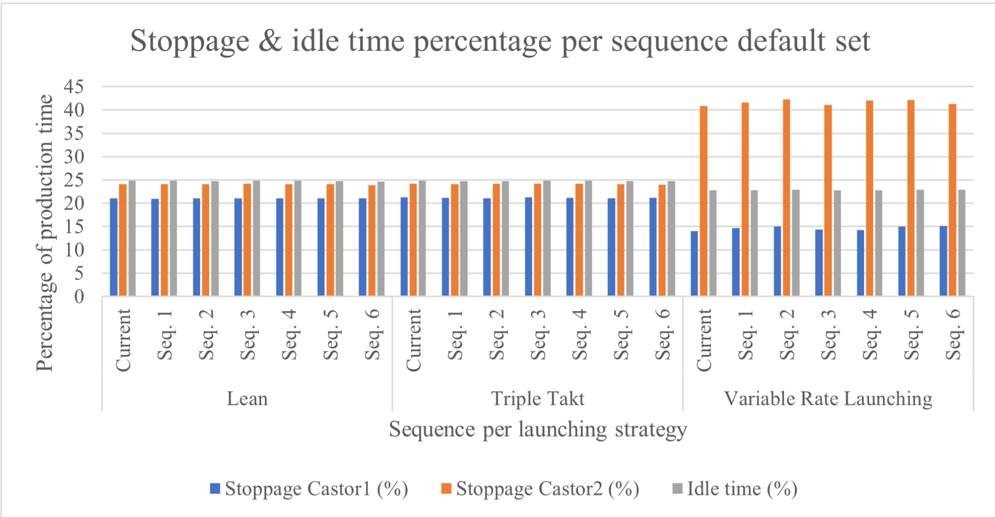


Figure 6.15: Average stop and idle times of the alternative solutions on default order mix experiments

Stochastic Experiments Sequence Performance

We measure the performance of sequences with specific characteristics for the long order set, as described in Section 5.2.3. Figure 6.16 shows the quality of our generated sequences, compared to the current sequence, for all three launching strategies. For the variable rate launching strategy, sequences result in the highest output when a long and a complex truck are sequenced consecutively. Here, it does not matter much if a long truck (seq. 7) or a complex truck (seq. 8) is planned first. The current sequence also includes consecutive pairs of long and complex trucks and therefore gives similar results. When long trucks are spread out evenly, without considering complex trucks (seq. 3) or with an even spread out of complex trucks (seq. 9), the sequences also result in a relatively high output. Sequences that only consider the complex processing times result in a relatively low output, especially if trucks with similar processing times are clustered together (seq. 4 & seq. 5). If the trucks are clustered based on their length, this give a lower output if the trucks are ordered from short to long (seq. 2), but a higher output if the trucks are ordered from long to short (seq. 3).

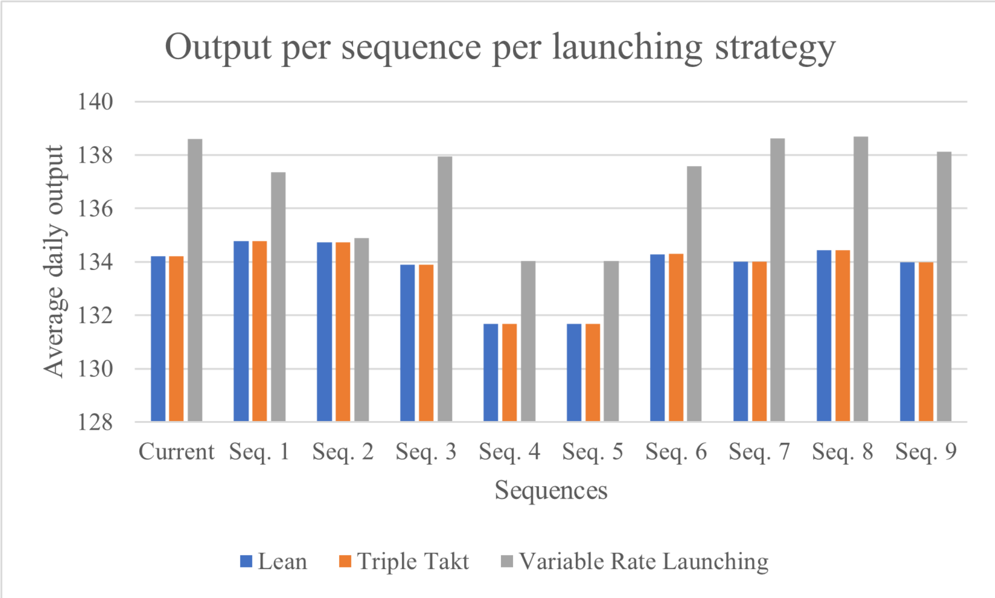


Figure 6.16: Behaviour of sequences with specific characteristics

We are able to explain that a combination of a long and a complex truck in a sequence results in good outcomes, since a long truck can provide more space (*buffer*) between two (complex) trucks, which leads to less stoppage time. Figure 6.17 illustrates this using two sub-sequences of three consecutive trucks, where blue blocks represent short trucks and red blocks represent long trucks. If the second truck is not finished yet at the end of station 33, a stoppage occurs. After this stoppage, the operators of station 33 can start at the following (third) truck. In the first situation, the truck is already halfway the station, which means that the operators can only use half of the station length before the truck leaves the station, which causes a line stoppage if the processing time of that truck is longer than half the takt time. In the second situation, this is not the case, because the long truck ensures that its following truck is only at the start of the work station when the operators start their tasks.

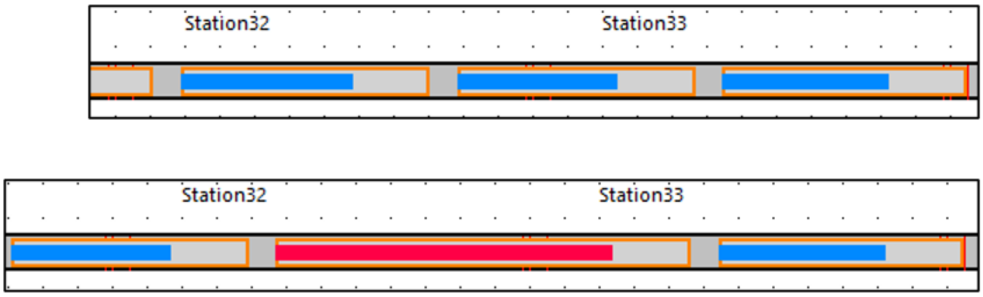


Figure 6.17: Illustration of three short trucks vs. two short and one long truck on two stations

Furthermore, it makes sense that complex trucks should not be clustered together, since it is more efficient when operators can start earlier on a complex truck, because they have time left after an *easy* truck. It is also understandable that an even spread of long trucks results in a high production output, since the already explained buffer effect then impacts more trucks. Besides, each long truck delays its following truck. Therefore, a sequence where trucks are ordered from long to short (seq. 2) yields still relatively good results, since the short trucks can later compensate for the delay. However, in a sequence where trucks are ordered from short to long (seq. 3), the short trucks cannot make up for the delay of these trucks and therefore lead to a decrease in output.

In the current (Lean) and triple takt launching strategy, all sequences yield relatively good results, except for the sequences where the trucks are clustered by similar processing times (seq. 4 & 5). In the current launching strategy, the short trucks are not able to compensate for the delay caused by the long trucks. In the triple takt launching strategy, only one shorter truck can compensate for a long truck, which does not differ much from the current strategy (see Figure 6.16). Therefore, the consideration of the lengths of the trucks for sequencing is less important with these launching strategies. However, it is interesting to see that if the trucks are ordered on their lengths (seq. 1 and 2), this gives slightly higher results than the current sequence. This could be explained by the fact that both consecutive short and long trucks result in less stoppages. Clustered short trucks can ensure that operators have space to start earlier on a complex truck, and long trucks give operators more available time to work, therefore both leading to fewer stoppages.

6.2.3 Sensitivity Analysis

We execute the 10 experiments from Table 5.5 for the five alternative scenarios. Figures 6.18 and 6.19 show the average daily output per experiment per scenario. For all alternative solutions, closed boundaries (exp. 2) and high technical stoppage percentages (exp. 7) decrease the average daily output the most, while broad open boundaries (exp. 3 & 4) and low technical stoppage percentages (exp. 5 & 6) increase the average daily output. The variance-to-mean ratio does not highly influence the outcomes (exp. 8 - 10). Changing the number of stations does influence the average daily output, but only if these stations are no bottleneck stations (exp. 11 vs. exp. 12). This sensitivity analysis also shows that the variable rate launching strategy is more robust than the current and triple takt strategies, because it is less responsive to the changes in the input parameters.

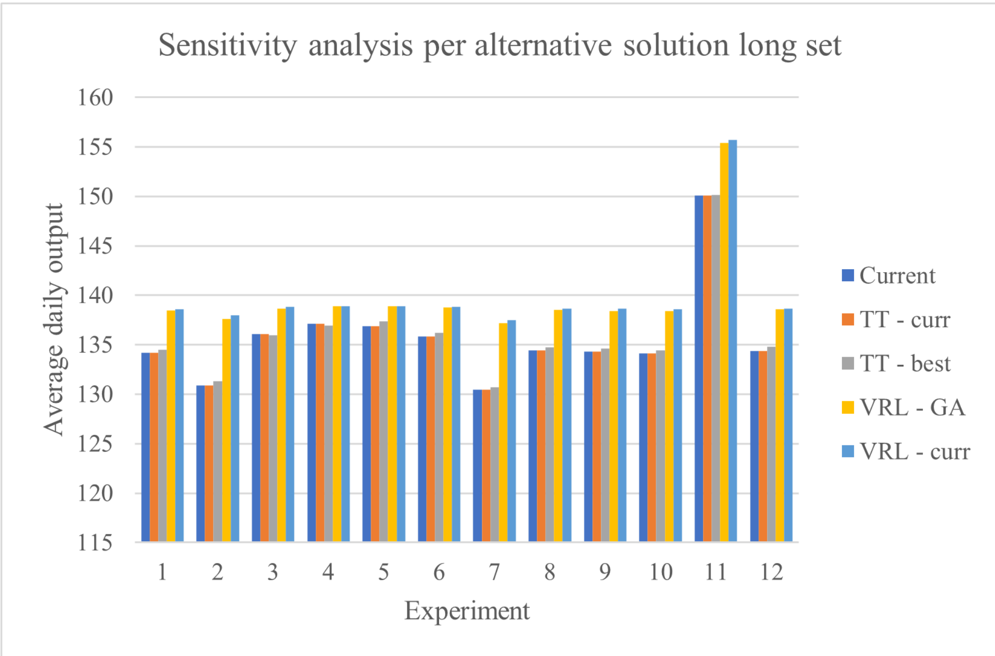


Figure 6.18: Sensitivity analysis for five scenarios with long order set

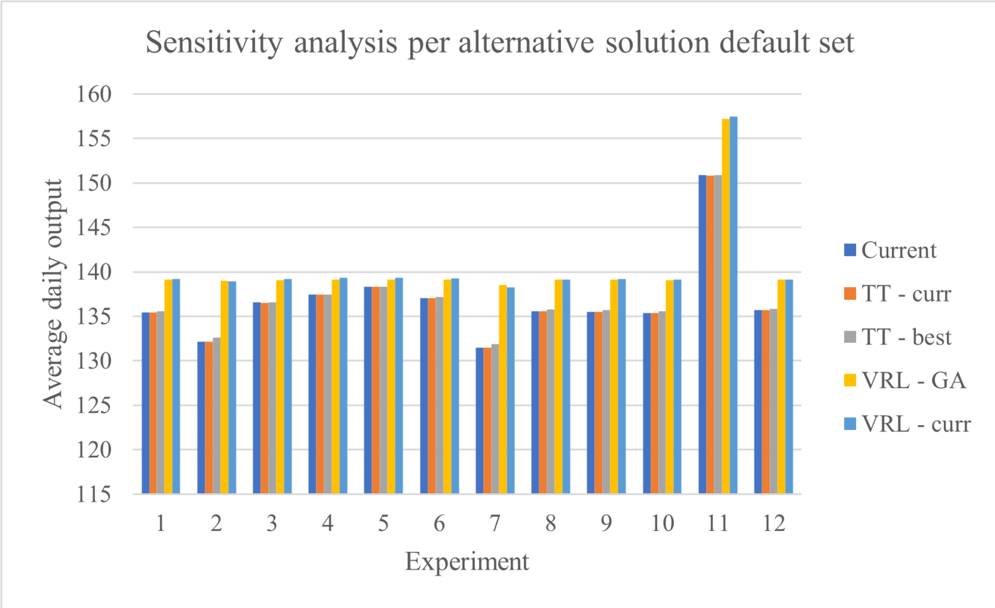


Figure 6.19: Sensitivity analysis for five scenarios with default order set

6.3 Conclusion

To conclude, our SE approach was not able to generate sequences for our mixed-model sequencing problem that resulted in high daily output. However, both our SH approach and our DES approach generated well-performing sequences. Our experiments show that the optimal repositioning of the light sensors leads to a slight increase of the production rate, and that the variable rate launching strategy (with a continuous measuring) is able to increase the current production rate by 3.2%, which is also the most robust strategy.

Chapter 7

Conclusion, Discussion & Recommendation

This chapter shows the conclusions of our research (Section 7.1), argues the limitations of our study and suggests further research directions (Section 7.2). Finally, Section 7.3 gives a recommendation of SPZ to optimize their assembly line.

7.1 Conclusion

In Chapter 1, we formulated the following research question for SPZ: *how can we maximize the output of a fixed speed assembly line with products with varying lengths?* We analysed the current situation of SPZ and their problems, and concluded that there were three improvement possibilities:

1. The measuring method, because the currently placed light sensors lead to output loss.
2. The launching strategy of the trucks to the carrier line with a fixed speed, because the current situation is not able to compensate the output loss caused by long trucks.
3. The sequencing approach, because the planning department currently only implicitly includes the length of the trucks (car-sequencing).

For the measuring method, we designed a linear program that places a given number of light sensors such that the total unnecessary measured distance is minimized. Our experiments showed that by adding four light sensors (9 in total) to the current situation and positioning them based on our discrete measuring location model, SPZ can increase its monthly production with 3 trucks (0.10 %). If the trucks are measured continuously, this leads to a total increase of 4 trucks (0.13 %). Both increases are statistically insignificant. If the repositioning of the light sensors does not lead to a high investment for SPZ, this repositioning is worth the investment for SPZ after a new introduction of trucks with a different length distribution.

We designed two alternative solutions for the launching strategy, the triple takt strategy that compensates the extra length of long trucks, and the variable rate launching strategy, which always launches trucks to the line when it is safe. Our experiments showed that the triple takt strategy only leads to a slight improvement compared to the current strategy, and mostly with a continuous measuring method. We also proved that the variable rate launching strategy is able to increase the daily production with 3.2%, if the lengths are continuously measured. Although this strategy increases the stoppage time at Castor2 significantly, the overall stoppage time (measured at Castor1) decreases, as well as the idle time. The sensitivity analysis showed that the variable rate launching strategy is also the most robust strategy.

We designed three genetic algorithm approaches to generate near-optimal sequences that explicitly consider the lengths and processing times of the trucks (mixed-model sequencing): a simplified evaluation (SE), a discrete-event simulation (DES) and a simheuristics (SH) approach. The SE approach was not able to evaluate the performance of a sequence sufficiently, because a complex system including multiple stations that result in full line stoppages is best modelled in a discrete-event simulation. The DES GA generated the best sequences using an order crossover and inversion, with a relative fitness reference. However, our SH approach led to similar outcomes, but with a faster run time compared to the GA in our DES model. For the variable rate launching strategy, the current sequence resulted in the highest daily production rate, but in general the sequences all gave similar outcomes.

We also analysed that sequences that result in a high production rate often include long trucks that are consecutive to *complex* trucks (with high processing times). An explanation of this is that long trucks can create a buffer of space on the assembly line such that the operators have more time to work on complex trucks in their work station.

To conclude, we can maximize the output on a fixed speed assembly line with products with varying lengths, using a variable rate launching strategy and a continuous measuring method. The current sequencing method already has a high performance, but for generating new sequences, a simheuristics approach performs well considering the balance between optimisation and evaluation.

7.2 Discussion

The validation of our DES model showed that the data on the processing times was not very accurate, since the processing times were often longer than in the current situation. Although we could still compare our alternative solutions to a benchmark of the current situation, since all alternatives included the same data, we recommend SPZ to invest in the quality of their data on the processing times.

The evaluation method included in the SE approach as we designed it, could not evaluate the performance of sequences well compared to the DES model. However, for further research we are interested in whether it would be possible to create a different evaluation method that gives similar results to a discrete-event simulation, but with a faster run time. Then, there could be more focus on the optimisation part, instead of the evaluation. If this is the case, the outcomes from the fitness calculation could be updated to resemble the outcomes for our DES model, by adding certain *penalties* based on characteristics of the sequence. These penalties could be determined with a linear regression model, where the dependent variables are the characteristics of a sequence that should forecast the independent variable, the gap between the outcomes of the two models. For the characteristics of sequences, we suggest using aspects such as:

1. The maximum number of consecutive long trucks ($>$ 12 m.).
2. The maximum number of consecutive short trucks ($<$ 10 m.).
3. The number of violations of the mixing rule of only one long truck per three trucks.
4. The maximum number of consecutive trucks with a high task time ($>$ 400 s.) per station.
5. The maximum, minimum, and standard deviation of the *cumulative compensation* values. For each sequence, we determine the difference between the takt time and launching interval between two consecutive trucks, the so-called *compensation values*. Then, we determine the cumulative compensation values per truck in the sequence.

The current sequence resulted in a higher daily production rate than the best GA generated sequence for the long order set, which suggests that the GA is not able to find good sequences. However, it could also mean that the current sequence is actually the optimal solution for this mixed-model sequencing problem, and since we never know if metaheuristics result in the optimal solution, we do not have to conclude that the performance of the GA in Plant Simulation is actually insufficient. For further research, we suggest that a DES software is used that is able to include solutions in the first generation of the GA, such that the current sequence is actually included in the GA from the beginning.

In the experiments and sensitivity analysis, we did not include a full-factorial analysis, which means that we did not analyse how parameters affect each other. For further research, we would suggest including this analysis, in order to understand specific characteristics of mixed-model assembly lines and their influence on the production rate better.

For further research, we recommend looking more into simheuristics, which is a promising approach for the complex balance of optimisation and evaluation. Our SH approach for the GA applied only one iteration of the methodology of Juan et al. (2015) (see Figure 3.12), but we suggest that further research applies multiple iterations. We are interested in the improvement per iteration and the characteristics of a good deterministic model in mixed-model sequencing problems.

Furthermore, we also recommend SPZ to investigate various operations research methods for optimizing online sequencing. The current sequencing method and our developed GA assume that all trucks in the order set are assembled in their planned assembly period, while design or logistical issues could occur that delay certain trucks. Online sequencing methods, that attempt to optimize a current sequence with changes, create more improvement possibilities.

7.3 Recommendation

We recommend SPZ to invest in a continuous measuring method, rather than investing in adding more light sensors and (re)positioning them, since this leads to the highest increase in production. In this way, SPZ is prepared for future carrier lines, that might not include the restriction of discrete carrier distances. If SPZ uses a continuous measuring method with their current carrier system, we advise them to use our discrete measuring location model to determine the 10 different carrier distances.

We recommend SPZ to invest in the quality of the data on their processing times. This would not only improve the results of this study, but also lead to a detailed bottleneck analysis, which could optimize the entire assembly line.

We recommend SPZ to research the possibility of variable rate launching more in detail, specifically regarding the impact on personnel and the current existing hard- and software. Although many engineers at SPZ think that variable rate launching leads to more full line stoppages, our research showed that this is not the case. However, this launching strategy needs a change in the way of thinking of management and personnel, and we recommend SPZ and specifically the *Scania Production System* department to analyse the change management that is needed for this new launching strategy. Management of SPZ should decide whether this investment is worth the production rate improvement of 3%. If this is not the case, the triple takt strategy would still improve the production rate slightly. The triple takt and the variable rate launching strategy both require similar hard- and software changes, so the triple takt strategy could also be a first step to the variable rate launching strategy.

We encourage SPZ to maintain their current car-sequencing GA, since our experiments showed that it has great performance, which suggests that only implicitly including lengths and processing times of trucks is sufficient. We recommend SPZ to include implicit rules based on our research, such as the consecutive sequencing of a long and a complex truck.

Next to our designed alternative solutions, the sensitivity analysis also showed other improvement possibilities for SPZ to increase the daily production rate, such as investing in open borders of the stations, and decreasing the technical stoppages. So, if changing the launching strategy is not possible, we recommend SPZ to research the bottlenecks that cause closed borders (such as fixed equipment) and continuing the root cause analysis of technical stoppages.

Bibliography

Abdel-Basset, M., Abdel-Fatah, L., & Sangaiah, A. K. (2018). Metaheuristic Algorithms: A Comprehensive Review. In A. K. Sangaiah, M. Sheng, & Z. Zhang (Eds.), *Computational intelligence for multimedia big data on the cloud with engineering applications* (pp. 185–231). Academic Press. https://doi.org/10.1016/B978-0-12-813314-9.00010-4

Abdullah Make, M. R., Mohd Fadzil, M. F. F., & Razali, M. M. (2017). A review of two-sided assembly line balancing problem. *The International Journal of Advanced Manufacturing Technology*, *89*(5), 1743–1763. https://doi.org/10.1007/S00170-016-9158-3

Alford, D., Sackett, P., & Nelder, G. (2000). Mass customisation — an automotive perspective. *International Journal of Production Economics*, *65*(1), 99–110. https://doi.org/10.1016/S0925-5273(99)00093-6

Ali, R. M., & Deif, A. M. (2014). “Dynamic Lean Assessment for Takt Time Implementation”. *Procedia CIRP*, *17*, 577–581. https://doi.org/10.1016/J.PROCIR.2014.01.128

Biele, A., & Manch, L. (2015). Using simulation to improve planning decisions in mixed-model assembly lines. *Winter Simulation Conference*. https://doi.org/10.1109/WSC.2015.7408328

Black, J. T. (2007). Design rules for implementing the Toyota Production System. *International Journal of Production Research*, *45*(16), 3639–3664. https://doi.org/10.1080/00207540701223469

Boysen, N., Fliedner, M., & Scholl, A. (2009). Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research*, *192*(2), 349–373. https://doi.org/10.1016/J.EJOR.2007.09.013

Boysen, N., Fliedner, M., & Scholl, A. (2010). Production planning of mixed-model assembly lines: overview and extensions. *Production Planning & Control*, *20*(5), 455–471. https://doi.org/10.1080/09537280903011626

Boysen, N., Schulze, P., & Scholl, A. (2021). Assembly line balancing: What happened in the last fifteen years? *European Journal of Operational Research*. https://doi.org/10.1016/J.EJOR.2021.11.043

Brammer, J., Lutz, B., & Neumann, D. (2021). Solving the mixed model sequencing problem with reinforcement learning and metaheuristics. *Computers & Industrial Engineering*, *162*, 107704. https://doi.org/10.1016/J.CIE.2021.107704

Cassandras, C. G., & Lafortune, S. (2008). Introduction to Discrete-Event Simulation. *Introduction to discrete event systems* (2nd, pp. 557–615). Springer. https://doi.org/10.1007/978-0-387-68612-7{_}10

Cho, H. S., Paik, C. H., Yoon, H. M., & Kim, H. G. (2005). A robust design of simulated annealing approach for mixed-model sequencing. *Computers & Industrial Engineering*, *48*(4), 753–764. https://doi.org/10.1016/J.CIE.2004.12.005

Daskin, M. S. (2008). What you should know about location modeling. *Naval Research Logistics (NRL)*, *55*(4), 283–294. https://doi.org/10.1002/NAV.20284

Dinesh, S., Gupta, V., Sethy, D., & Guptaz, V. (2005). Application of value stream mapping for lean operations and cycle time reduction: an Indian case study. *Production Planning & Control*, *16*(1), 44–59. https://doi.org/10.1080/09537280512331325281

Downey, B. S., & Leonard, M. S. (2007). Assembly line with flexible work-force. *International Journal of Production Research*, *30*(3), 469–483. https://doi.org/10.1080/00207549208942906

Eiben, A., & Smith, J. (2015). *Introduction to Evolutionary Computing* (2nd). Springer. https://doi.org/10.1007/978-3-662-44874-8

Faccio, M., Gamberi, M., & Bortolini, M. (2015). Hierarchical approach for paced mixed-model assembly line balancing and sequencing with jolly operators. *International Journal of Production Research*, *54*(3), 761–777. https://doi.org/10.1080/00207543.2015.1059965

Fang, J., Zhao, L., Fransoo, J. C., & Van Woensel, T. (2013). Sourcing strategies in supply risk management: An approximate dynamic programming approach. *Computers and Operations Research*, *40*(5), 1371–1382. https://doi.org/10.1016/j.cor.2012.08.016

Fattahi, P., & Salehi, M. (2009). Sequencing the mixed-model assembly line to minimize the total utility and idle costs with variable launching interval. *The International Journal of Advanced Manufacturing Technology 2009 45:9*, *45*(9), 987–998. https://doi.org/10.1007/S00170-009-2020-0

Gilmore, J., & Pine II, B. (1997). The four faces of mass customization. *Harvard Business Review*, *75*(1).

Goldberg, D., & Lingle, R. (1985). Alleles, Loci and the Traveling Salesman Problem. *1st International Conference on Genetic Algorithms and Their Applications*, 154–159.

Hatami, S., Calvet, L., Fernández-Viagas, V., Framiñán, J. M., & Juan, A. A. (2018). A simheuristic algorithm to set up starting times in the stochastic parallel flowshop problem. *Simulation Modelling Practice and Theory*, *86*, 55–71. https://doi.org/10.1016/J.SIMPAT.2018.04.005

Hemig, C., Rieck, J., & Zimmermann, J. (2014). Integrated production and staff planning for heterogeneous, parallel assembly lines: an application in the automotive industry. *International Journal of Production Research*, *52*(13), 3966–3985. https://doi.org/10.1080/00207543.2013.828174

Hong, L. J., & Nelson, B. L. (2009). A brief introduction to optimization via simulation. *Proceedings - Winter Simulation Conference*, 75–85. https://doi.org/10.1109/WSC.2009.5429321

Huchzermeier, A., Mönch, T., & Bebersdorf, P. (2020). The fendt VarioTakt: Revolutionizing mixed-model assembly line production. *INFORMS Transactions on Education*, *20*(3), 141–153. https://doi.org/10.1287/ITED.2019.0224CA

Jawahar, N., Ponnambalam, S. G., Sivakumar, K., & Thangadurai, V. (2014). Heuristics for multiobjective optimization of two-sided assembly line systems. *The Scientific World Journal*, *2014*. https://doi.org/10.1155/2014/458959

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M., & Figueira, G. (2015). A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. *Operations Research Perspectives*, *2*, 62–72. https://doi.org/10.1016/j.orp.2015.03.001

Juurlink, J. (2021). Line layout and assembly line balance proposal to reduce output loss due to long trucks at Scania.

Kampker, A., Kawollek, S., Marquardt, F., & Krummhaar, M. (2020). Potential of Hybrid Assembly Structures in Automotive Industry. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.3848756

Kern, W., Rusitschka, F., Kopytynski, W., Keckl, S., & Bauernhansl, T. (2015). Alternatives to assembly line production in the automotive industry. *23rd International Conference for Production Research, ICPR 2015*.

Kilic, H. S., & Durmusoglu, M. B. (2015). Advances in assembly line parts feeding policies: A literature review. *Assembly Automation*, *35*(1), 57–68. https://doi.org/10.1108/AA-05-2014-047/FULL/XML

Kumar, R., & Jyotishree. (2012). Blending Roulette Wheel Selection & Rank Selection in Genetic Algorithms. *International Journal of Machine Learning and Computing*, *2*(4).

Law, A. M. (2015). *Simulation modeling and analyis* (5th, Inter). McGraw-Hill Education.

Lv, Y., Zhang, J., & Qin, W. (2019). Simulation-based production analysis of mixed-model assembly lines with uncertain processing times. *Journal of Simulation*, *13*(1), 44–54. https://doi.org/10.1080/17477778.2018.1436419

Martignago, M., Battaïa, O., & Battini, D. (2017). Workforce management in manual assembly lines of large products: a case study. *IFAC-PapersOnLine*, *50*(1), 6906–6911. https://doi.org/10.1016/J.IFACOL.2017.08.1215

Mohseni-Darabi, S., Mahmoodjanloo, M., & Baboli, A. (2021). Sequencing Problem in a Paced Mixed-Model Assembly Line with Multiple Flexible Operators. *IEEE 8th International Conference on Industrial Engineering and Applications*, 290–295. https://doi.org/10.1109/ICIEA52957.2021.9436689

Mönch, T., Huchzermeier, A., & Bebersdorf, P. (2020). Variable takt time groups and workload equilibrium. *International Journal of Production Research*. https://doi.org/10.1080/00207543.2020.1864836

Mönch, T., Huchzermeier, A., & Bebersdorf, P. (2021). Variable takt times in mixed-model assembly line balancing with random customisation. *International Journal of Production Research*, *59*(15), 4670–4689. https://doi.org/10.1080/00207543.2020.1769874

Oudhuis, M., & Tengblad, S. (2020). The viability of the Scandinavian work-life model and the impact of lean production: The case of Scania: *Economic and Industrial Democracy*. https://doi.org/10.1177/0143831X20939137

Prombanpong, S., Dumkum, C., & Satranonda, E. (2010). A fixed rate launching of mixed- model car sequencing in the multiple assembly lines. *IEEE International Conference on Industrial Engineering and Engineering Management*, 1266–1270. https://doi.org/10.1109/IEEM.2010.5674378

Rane, A. B., Sudhakar, D. S., Sunnapwar, V. K., & Rane, S. (2015). Improving the performance of assembly line: Review with case study. *International Conference on Nascent Technologies in the Engineering Field*. https://doi.org/10.1109/ICNTE.2015.7029913

Rodríguez Parral, A. V., & López Pérez, J. F. (2020). Optimization Model for Production Scheduling Requirements Applied on Heavy Truck Assembly Lines. *Computación y Sistemas*, *24*(3), 999–1007. https://doi.org/10.13053/CYS-24-3-3152

Saif, U., Guan, Z., Liu, W., Wang, B., & Zhang, C. (2014). Multi-objective artificial bee colony algorithm for simultaneous sequencing and balancing of mixed model assembly line. *The International Journal of Advanced Manufacturing Technology*, *75*(9-12), 1809–1827. https://doi.org/10.1007/s00170-014-6153-4

Salehi, M., Fattahi, P., Roshani, A., & Zahiri, J. (2012). Multi-criteria sequencing problem in mixed-model synchronous assembly lines. *The International Journal of Advanced Manufacturing Technology 2012 67:1*, *67*(1), 983–993. https://doi.org/10.1007/S00170-012-4542-0

Sawik, T. (2004). Loading and scheduling of a flexible assembly system by mixed integer programming. *European Journal of Operational Research*, *154*(1), 1–19. https://doi.org/10.1016/S0377-2217(02)00795-6

Scania CV AB. (n.d.). Scania in brief. https://www.scania.com/group/en/home/about-scania/scania-in-brief.html

Scania Production Zwolle. (2022). Strategisch Plan MZ 2022.

Schmeiser, B. (1982). Batch Size Effects in the Analysis of Simulation Output. *Operations Research*, *30*(3), 556–568. https://doi.org/10.1287/OPRE.30.3.556

Serkan Akgündüz, O., & Tunali, S. (2010). A review of the current applications of genetic algorithms in mixed-model assembly line sequencing. *International Journal of Production Research*, *49*(15), 4483–4503. https://doi.org/10.1080/00207543.2010.495085

Shi, X., Cui, W., Zhu, P., & Yang, Y. (2021). Research on Automobile Assembly Line Optimization Based on Industrial Engineering Technology and Machine Learning Algorithm. *Machine Learning for Energy Efficient Wireless Communications and Mobile Computing*, *2021*(Special Issue). https://doi.org/10.1155/2021/2658090

Suh, N. P., Cochran, D. S., & Lima, P. C. (1998). Manufacturing System Design. *CIRP Annals*, *47*(2), 627–639. https://doi.org/10.1016/S0007-8506(07)63245-4

Svensson Harari, N., Osterman, C., Bruch, J., & Jackson, M. (2014). Flexibility in Lean Mixed Model Assembly Lines. *IFIP Advances in Information and Communication Technology*, *440*(PART 3), 224–231. https://doi.org/10.1007/978-3-662-44733-8{_}28

Tiacci, L. (2012). Event and object oriented simulation to fast evaluate operational objectives of mixed model assembly lines problems. *Simulation Modelling Practice and Theory*, *24*, 35–48. https://doi.org/10.1016/J.SIMPAT.2012.01.004

Tsai, L. H. (1995). Mixed-Model Sequencing to Minimize Utility Work and the Risk of Conveyor Stoppage. *Management Science*, *41*(3), 485–495. https://doi.org/10.1287/MNSC.41.3.485

Vanguri, S., Hill, T. W., & Greenwood, A. G. (2006). Optimizing Flow Shop Sequencing Through Simulation Optimization Using Evolutionary Methods. *IIE Annual Conference - Proceedings*.

Wang, L.-F., & Shi, L.-Y. (2013). Simulation Optimization: A Review on Theory and Applications. *Acta Automatica Sinica*, *39*(11), 1957–1968. https://doi.org/10.1016/S1874-1029(13)60081-6

Welch, P. (1983). The Statistical Analysis of Simulation Results. In S. Lavenburg (Ed.), *Computer performance modeling handbook* (pp. 268–328). Academic Press.

Wirabhuana, A., Haron, H., & Imtihan, M. R. (2008). Simulation and re-engineering of truck assembly line. *Proceedings - 2nd Asia International Conference on Modelling and Simulation, AMS 2008*, 783–787. https://doi.org/10.1109/AMS.2008.82

Xiaobo, Z., & Ohno, K. (2000). Properties of a sequencing problem for a mixed model assembly line with conveyor stoppages. *European Journal of Operational Research*, *124*(3), 560–570. https://doi.org/10.1016/S0377-2217(99)00198-8

Yilmaz, H., & Yilmaz, M. (2019). A mathematical model and tabu search algorithm for multi-manned assembly line balancing problems with assignment restrictions. *Engineering Optimization*, *52*(5), 856–874. https://doi.org/10.1080/0305215X.2019.1618288

Zhang, W., Hou, L., & Jiao, R. J. (2021). Dynamic takt time decisions for paced assembly lines balancing and sequencing considering highly mixed-model production: An improved artificial bee colony optimization approach. *Computers & Industrial Engineering*, *161*, 107616. https://doi.org/10.1016/J.CIE.2021.107616

Zhao, X., Ohno, K., & Lau, H. S. (2004). A balancing problem for mixed model assembly lines with a paced moving conveyor. *Naval Research Logistics (NRL)*, *51*(3), 446–464. https://doi.org/10.1002/NAV.10116

Appendix A

Results of Run Characteristics of DES Model Calculations

Our simulation can be defined as a non-terminating simulation, since there is no natural event that specifies the end of a simulation run. To overcome this, we use the batch-means method (Schmeiser, 1982), which means that we have one run, with one warm-up period, after which we repeat the same order batch multiple times. To determine the warm-up period and number of runs, we run the simulation model for 200 days, for the long order set, since this set has a higher variability. We run the simulation model for multiple scenarios:

1. the current situation (Lean launching, current sequencing, measuring by current light sensors),
2. scenarios with variable rate launching, with the current and three random generated sequences, with continuous truck length measuring and current situation parameters, and
3. a scenario with variable rate launching, the current sequence, and high variability (higher stoppage percentages and a higher variance-to-mean ratio).

Figure A.1 shows the daily production rate for all these scenarios. All scenarios have a warm-up period and show a steady-state behaviour afterwards. If we use Welch's graphical method to determine the warm-up period for these scenarios (Welch, 1983), the warm-up period is dependent on the chosen window (see Figure A.2). Therefore, we decided to use for a warm-up period of 1 day, because only the first day influences the first moving averages of Welch's method, this is visible in Figure A.1.

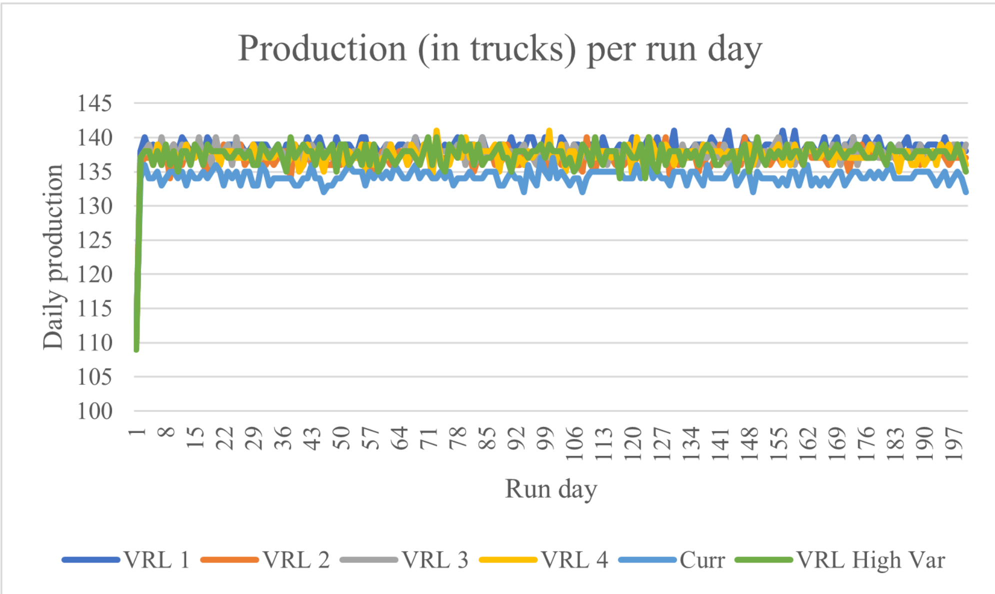


Figure A.1: Average production rate per run day for 6 scenarios

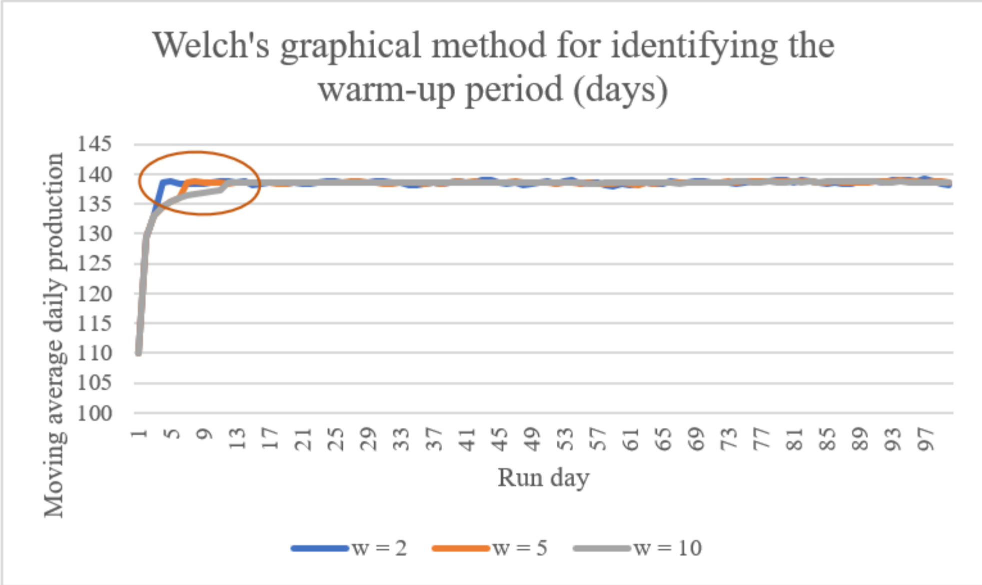


Figure A.2: Welch's graphical method for identifying the warm-up period (in days) for multiple windows, for scenario VRL 1

We choose a run length of 16 days, such that it is significantly larger than the warm-up period. We use the statistical analysis approach by Law (2015) to determine the number of replications that is required such that the width of the 95%-confidence interval relative to the mean is sufficiently small ($< 5\%$). Each replication is independent. For all six scenarios, 2 replications are sufficient for the average daily output KPI. For the other KPIs (stoppage time and idle time percentages), a maximum of 4 replications is required based on all scenarios. Since we only consider six different scenarios, we add one extra replication in order to have a sufficiently small width of the confidence interval in all scenarios. Therefore, we use 3 replications for the GA, which is only focused on the average daily output KPI, and 5 replications for the experiments where we consider all KPIs.

Appendix B

Significance of Solutions compared to the Current Situation

For all experiments that compare the current situation with our alternative solutions for launching strategies, light sensors and sequences, we calculate if the difference between the alternative solution and the current situation is significant. For this, we use a 95%-confidence interval of the difference of the daily production rate, which shows that the difference between two solutions is 95% significant if it does not include 0. Tables B.1, B.2 and B.3 present all confidence intervals, which are orange if the difference is insignificant, and green if the difference is significant.

Table B.1 Confidence intervals of light sensor experiments

Solution 1			**Solution 2**			**95%-Confidence Interval**	
Measuring	**Launching**	**Sequencing**	**Measuring**	**Launching**	**Sequencing**	**Left**	**Right**
Current	Lean	Current	5 opt.	Lean	Current	-0.55	0.30
Current	Lean	Current	10 opt.	Lean	Current	-0.74	0.11
Current	Lean	Current	Continuous	Lean	Current	-0.54	0.16
Current	TT	Current	5 opt.	Lean	Current	-0.55	0.30
Current	TT	Current	10 opt.	Lean	Current	-0.74	0.11
Current	TT	Current	Continuous	Lean	Current	-0.91	0.04
5 opt.	VRL	Current	10 opt.	VRL	Current	-0.75	0.37
5 opt.	VRL	Current	Continuous	VRL	Current	-1.01	-0.24
Current	Lean	Current	10 opt.	TT	Current	-0.74	0.11
Current	Lean	Current	Continuous	TT	Current	-0.91	0.04
Current	Lean	Current	10 opt.	VRL	Current	-4.71	-2.79
Current	Lean	Current	Continuous	VRL	Current	-4.84	-3.54

Table B.2 Confidence intervals of sequence experiments of default order set

Solution 1			**Solution 2**			**95%-Confidence Interval**	
Measuring	**Launching**	**Sequencing**	**Measuring**	**Launching**	**Sequencing**	**Left**	**Right**
Cont.	Lean	Current	Cont.	TT	Current	-0.39	0.39
Cont.	Lean	Current	Cont.	TT	GA - 1	-0.60	0.72
Cont.	Lean	Current	Cont.	TT	GA - 2	-0.81	0.43
Cont.	Lean	Current	Cont.	TT	GA - 3	-0.67	0.42
Cont.	Lean	Current	Cont.	TT	GA - 4	-0.67	0.42
Cont.	Lean	Current	Cont.	TT	GA - 5	-0.59	0.47
Cont.	Lean	Current	Cont.	TT	GA - 6	-0.66	0.16
Cont.	Lean	Current	Cont.	VRL	Current	-4.32	-3.18
Cont.	Lean	Current	Cont.	VRL	GA - 6	-4.05	-2.95
Cont.	Lean	Current	Cont.	Lean	GA - 6	-0.64	0.39
Cont.	TT	Current	Cont.	TT	GA - 2	-0.70	0.45
Cont.	VRL	Current	Cont.	VRL	GA - 6	-0.59	1.09

Table B.3 Confidence intervals of sequence experiments of long order set

Solution 1			**Solution 2**			**95%-Confidence Interval**	
Measuring	**Launching**	**Sequencing**	**Measuring**	**Launching**	**Sequencing**	**Left**	**Right**
Cont.	Lean	Current	Cont.	TT	Current	-0.76	0.01
Cont.	Lean	Current	Cont.	TT	GA - 1	-1.26	-0.49
Cont.	Lean	Current	Cont.	TT	GA - 2	-1.05	-0.20
Cont.	Lean	Current	Cont.	TT	GA - 3	-1.06	-0.44
Cont.	Lean	Current	Cont.	TT	GA - 4	-0.98	-0.02
Cont.	Lean	Current	Cont.	TT	GA - 5	-1.01	-0.24
Cont.	Lean	Current	Cont.	TT	GA - 6	-0.99	0.11
Cont.	Lean	Current	Cont.	VRL	Current	-4.62	-3.38
Cont.	Lean	Current	Cont.	VRL	GA - 6	-4.48	-3.52
Cont.	Lean	Current	Cont.	Lean	GA - 2	-0.63	0.26
Cont.	TT	Current	Cont.	TT	GA - 3	-0.76	0.01
Cont.	VRL	Current	Cont.	VRL	GA - 6	-0.67	0.67

Appendix C

Confidential Information

[Restricted]