Chair of Industrial Logistics

Master's Thesis

Simulation-based Bottleneck Identification in a Job-shop with Changing Product Mixture and Production Volumes

Dionysius Viehhauser, BSc

February 2023

Affidavit

MONTANUNIVERSITÄT LEOBEN
www.unileoben.ac.at

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich diese Arbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt, und mich auch sonst keiner unerlaubten Hilfsmittel bedient habe.

Ich erkläre, dass ich die Richtlinien des Senats der Montanuniversität Leoben zu "Gute wissenschaftliche Praxis" gelesen, verstanden und befolgt habe.

Weiters erkläre ich, dass die elektronische und gedruckte Version der eingereichten wissenschaftlichen Abschlussarbeit formal und inhaltlich identisch sind.

Datum 14.02.2023

Unterschrift Verfasser/in
Dionysius Viehhauser

Dedication

<u>Thanks and appreciation to all out there making years of study a lifetime experience</u>

Embracing gratitude to my love and family: Anna, mum & dad thanks for your patience & constant support that enabled this journey from begin on. Thank you for being there when needed, including the support from Lukas.

Deep gratefulness to study companion, friend and thesis advisor Alex.

Thankful for essential time care, possibility basis and expert contribution from Martin, Dorit, Leon and Ivalina, including understanding appreciation from Marius and Sebastian.

Grateful for shapeable study possibilities enabled by Manuel and Helmut Z.; and Paul OL. for highlighting the exciting properties of mathematical and software opportunities.

Cheers for early focus orientation in logistics, automotive, processes and storytelling Carsten.

Thanks for mostly harmonising :P living conditions with Deisl, Nikozlaus, Raphi and Raphael, making the best surrounding for a reasonable student life possible.

Spending unforgettable days and evenings with Shelly, Schnädi, Nox, Rocco, Schurl, Bade, Priescht, Konsti, Dousinger, KJ, Dani, Aufstieg, Knieriem, Calle, Tom, Bambi, Willi, Bobby, Momo, Percht, Stierli, Cravallo, Clemens, Heinrich, Eugene, Hans, Marcel, Perkeo, Bierlein, Ninive, Fruchtl, Lois, Rama, Blanki, Luggi & Templer.

Appreciated study buddies Elli, Martin, Peter & Peter, Andi, Janisch, Eva & Toni.

Best basis friendships with Fabian, Voith, Alex, Hagi, Thomas, Hepp, Dani & Sebastian.

Skiing & crazy bros Dominik, Yannick, Birgit, Carsten, Alex & Dave.

Alumni love to Mäscht, Geri, Tobi, Lukas, Schneider, Markus, Kattinger, Maxi & Twins.

By mentioning a *Chinese quote 'Not the wind, but the sail determines the direction.'*, the author wants to thank you all for accompanying this journey.

Dank und Anerkennung to have made it this far together with you.

*Ein letztes montan**studentisches** Glück Auf!*

Abstract

Considering a production environment with different fabrication technologies, frequently called job-shop, many interdependencies exist that impact the final lead time of a production order and the overall system performance. Bottlenecks represent a crucial role regarding output by influencing the amount of present work in process (WIP), passing time an order spends within a system and the utilisation of other resources. Those influence factors underline, that bottlenecks play a major role for successful operations in manufacturing. Therefore, this thesis examines the possibilities to determine bottlenecks of a dynamic fabrications area for historian and future scenarios, additionally evaluating resulting characteristics.

At the beginning, theoretical fundamentals are discussed related to the determination of appropriate key performance indicators and to sorts of bottleneck identification methods. Subsequently, simulation basics are explained, and the implications of all mentioned principles get summarised.

In order to test the proposed procedures, a generic Python framework is introduced. A discrete-event simulation (DES) model is realised by using the open-source library salabim. The configurable virtual model is applied as a case study to a job-shop of an industry partner including 43 workplaces, handling 9 different fabrication technologies of the metal processing industry. Moreover, considered parameters defining such a system are implemented like stochastic distributed workplace breakdowns, resulting WIP and lead time of production orders, defined shift times, weekend overtimes and alternative workplaces. The complete model development is accompanied by constant verification and validation (V&V) measures. Finally, the suggested approach is successfully used to identify static and shifting bottlenecks of a dynamic job-shop environment for past and future scenarios including a different product mix and changing production volumes.

Kurzfassung

In einer fertigungsinsel-organisierten Produktionsumgebung mit unterschiedlichen Fertigungsverfahren und Technologien, bestehen viele Interdependenzen, die sich auf die Durchlaufzeit eines Produktionsauftrags und die Gesamtleistung des Systems auswirken. Engpässe spielen eine entscheidende Rolle für die Leistung eines Produktionssystems, da sie die Menge des Umlaufbestandes (WIP), die Durchlaufzeit von Aufträgen durch das Fertigungssystem und die Auslastung anderer Ressourcen beeinflussen. Diese Faktoren unterstreichen, dass Engpässe eine entscheidende Rolle für einen erfolgreichen Produktionsprozess darstellen. In dieser Arbeit werden daher die Möglichkeiten zur Ermittlung von Engpässen in einem dynamischen Produktionsbereich für vergangene und zukünftige Szenarien untersucht und die daraus resultierenden Kenngrößen bewertet.

Zu Beginn werden theoretische Grundlagen zur Ermittlung geeigneter Kennzahlen und zu Arten von Engpassidentifikationsmethoden diskutiert. Anschließend werden Simulationsgrundlagen erläutert und die Implikationen aller genannten Prinzipien abgeleitet.

Um das vorgeschlagene Verfahren zu testen, wird ein generisches Python-Framework vorgestellt. Ein ereignis-diskretes Simulationsmodell wird mit Hilfe der Open-Source Bibliothek salabim realisiert. Das konfigurierbare virtuelle Modell wird im Rahmen einer Fallstudie auf eine Werkstattfertigung eines Industriepartners angewandt. Diese umfasst 43 Arbeitsplätze inklusive 9 verschiedener Fertigungstechnologien der metallverarbeitenden Industrie. Darüber hinaus werden Parameter, die ein solches System definieren, wie beispielsweise stochastisch verteilte Maschinenausfälle, resultierende WIP und Durchlaufzeiten von Fertigungsaufträgen, definierte Schichtzeiten, Wochenendüberstunden und Alternativarbeitsplätze berücksichtigt. Die gesamte Modellentwicklung wird durch ständige Verifikations- und Validierungsmaßnahmen (V&V) begleitet. Schließlich wird die vorgeschlagene Herangehensweise erfolgreich zur Identifizierung von statischen und dynamischen Engpässen in einer dynamischen Werkstattfertigung für vergangene und zukünftige Szenarien einschließlich eines unterschiedlichen Produktmixes und sich ändernder Produktionsmengen angewendet.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AB … agent-based (modelling)

dd … 2-digit day (e.g. 22)

DBR … drum-buffer-rope

DES … discrete-event simulation

DF … DataFrame (pandas)

DS … dynamic systems

DT … digital twin

ERP … enterprise resource planning

FEL … future event list

IDE … integrated development environment

IQR … interquartile range

KPI … key performance indicator

JSSP … job-shop scheduling problem

MAE … mean absolute error

mm … 2-digit month (e.g. 04)

MRP … materials requirement planning

(M)TBF … (mean-) time-between-failure

MTM … methods time measurement

MTO … make-to-order

(M)TTR … (mean-) time-to-repair

MU … monetary unit

PCS … production control system

RMSE … root mean square error

SD … system dynamics

TOC … Theory of Constraints

WIP … work in process

WPL … workplace

VSD … value stream design

VSM … value stream map

V&V … verification and validation

YYYY … 4-digit year (e.g. 2022)

1 Introduction

Starting in the 1970s, computer-aided software applications were integrated into production planning processes, in order to meet higher customer and governmental requirements by making processes more efficient and reducing lead times, delays, disruptions or costs. Higher competitiveness between companies additionally increased software support, supplementarily enabled by higher computing power making algorithms more efficient and capable.[1]

Nowadays, industrial manufacturers strive towards smart manufacturing, expecting to gain increased productivity as well as reduced overall costs and energy consumption. Especially recent years showed how critical it is that factories adapt quickly while facing volatile material availability, shorter product lifecycles and changing customer demands.[2]

Within the era of the fourth industrial revolution, an interconnected manufacturing shopfloor represents a core element enabled by digitalised processes, workplaces and additional process participants. One main expectation to strive for this current megatrend is the achievement of decision support and to overlook the complexity of interdependent procedures that humans developed in the last centuries. A currently emerging simulation paradigm that promises to solve such challenges is called digital twin (DT) nowadays, which gets increasing interest in the industrial and scientific environment in recent years. An accurate virtual model represents the basis to model a real system, which allows to evaluate various scenarios. A fully deployed DT is characterised by further aspects, varying by considered authors- simply caused by the reason that there exists no common definition. Tao and Kritzinger see it as crucial that there exists a bi-directional real-time data exchange between the real and virtual entity, resulting in automatic data flow and changes between the real and virtual entities.[3]

This thesis won’t focus on different digital twin definitions and approaches, but rather develop a virtual model of the material flow in a workshop production environment. This model should represent the basis for further use cases, developing a smart factory step by step.

Research has demonstrated that simulating production systems contains big potentials in improving processes by increasing production capacity, decreasing WIP or balance production lines. Besides other simulation techniques, discrete-event simulation (DES) is one of the most frequently used to model, analyse, simulate, optimise and visualise manufacturing processes, material flows and logistics activities.[4]

Event-oriented perspective of discrete processes in general has the advantage to model the dynamic behaviour of a system with high precision. Using DES, all events happening

---

[1] cf. Chen, X.; Voigt, T. (2020); Kulezak, M. (2022); Fabri, M. et al. (2022), p. 204 pp.

[2] cf. Friederich, J. et al. (2022), p. 1 pp.

[3] cf. Kritzinger, W. et al. (2018), p. 1016 pp.; Friederich, J. et al. (2022), p. 1; Kaiblinger, A.; Woschank, M. (2022), p. 1 pp.; Zhang, H. et al. (2022), p. 417 p.;

[4] cf. Murphy, A. et al. (2020), p. 2 p.; Huynh, B. H. et al. (2020), p. 16 p.; Guo, H. et al. (2021), p. 65.;

in a process are simulated, following a routine dependent on the type of currently considered event. This involves the calculation of a new system state resulting by the occurred event, the planning of new in the future happening events as also carrying out statistical evaluations.[5]

Found research mainly focuses on assembly line simulations with a pre-defined tact time, limited in-line-buffers and storage of semi-finished parts. This thesis will handle a job-shop production environment with different technology workshops and buffers between them. As part of the analysis, the effect of a changing product mix and production volumes in general on resulting bottlenecks, WIP, lead time and resource utilisation is determined.

1.1 Initial Situation and Problem Statement

In order to effectively schedule planned orders to resources, the determination of resulting bottlenecks based on current production orders is key for a successfully balanced shopfloor. Ensuring transparency and knowing about limiting resources allows to manage inventory levels and WIP, equipment utilisation and order lead times in an efficient way. That’s why this thesis will focus on bottleneck identification and evaluation regarding a changing product mix and production volume. The gained knowledge should be utilised by providing a discrete-event simulation model framework, that enables the determination of bottlenecks in a generic manner.

Value stream mapping (VSM) is known as a common process to map down the material and information flow and identify bottlenecks in production regarding one evaluated shopfloor status. The approach to design a DES model of an exemplary production environment should offer the possibility to objectively analyse different scenarios that might be present in the future. In contrary to VSM calculations bottlenecks will not be considered on a mean basis but analysed by a more detailed methodology.[6]

That involves the examination of detailed queue waiting times in front of workplaces, stochastic distributions regarding machine downtimes and necessary overtimes of workplaces. Such circumstances lead to shifting bottlenecks resulting from a dynamic system behaviour similar to reality which cannot be identified by applying a static determination method like VSM.[7]

1.2 Objective of Thesis and Research Issue

The goal of the thesis is to setup a configurable virtual model for a job-shop production environment in order to identify bottlenecks for current or future production scenarios regarding existing workplaces. Based on the resulting bottlenecks in future scenarios including different product mix and production volume, action possibilities will be evaluated. In the empirical part of the thesis a discrete-event simulation model is

[5] cf. Hedtstück, U. (2013), p. 22 p.; Law, A. M. (2013), p. 6 pp.
[6] cf. Roser, C.; Shook, J. (2021), p. 415 p.
[7] cf. Roser, C. et al. (2015), p. 3.

validated by means of a case study at a job-shop of an international technology and mechanical engineering industry partner providing crane and lifting solutions.

This simulation-based approach should determine the effect on key performance indicators, such as lead time, WIP and resource utilisation, based on an historical data set as also future scenarios from the industry partner.

These intentions lead to the following research issue: **How can a simulation approach be used in order to identify bottlenecks in a dynamic job-shop production environment with changing product mixture and production volume?**

1.3 Structure of Thesis

This thesis is built on an explorative literature research regarding production bottleneck detection methods, logistics key performance indicators and simulation in the production environment. Based on a methodical derivation of the literature review, this work is split into 4 main chapters. The **introduction** discusses the initial situation and clarifies the research question. Part 2 focusses on **theoretical fundamentals** regarding used and common key performance indicators referring to the performance of logistics and production systems. Furthermore, bottleneck identification methods are explained, and their strengths and weaknesses stated. As this thesis concentrates on a simulation-based technique to determine static and shifting bottlenecks in a job-shop with changing product mix and production volume, basics concerning simulation of production systems are described in more detail. The second chapter is closed by summarising the implications of the mentioned theoretical inputs.

The third part **empirical exploration** derives the gained knowledge of the theoretical inputs and introduces the development, verification and validation of a generic DES framework. The designed virtual model is applied as a case study to a job-shop of an industry partner. First, the handled production system is described as conceptual model, followed by a detailed description of the performed data collection and preparation process. As formal model, a process diagram is presented which is then transformed into an executable model by using the Python open-source library salabim. The complete model development is accompanied by constant verification and validation (V&V) measures. Lastly, this model is utilised by evaluating bottlenecks resulting from future scenarios including a different product mix and production volumes.

In the end, part 4 covers a **final consideration** of the performed thesis by summarising the results and giving an outlook into further scientific activities.

2 Theoretical Fundamentals

This second section presents reviewed scientific literature which represents the basis of this thesis. First, production system principles, basic terminologies used in the context of manufacturing and logistics are defined. Later, key performance indicators used in logistics and production are defined. Furthermore, different methods regarding bottleneck identification are mentioned including their advantages and limitations. Since this thesis focuses on simulation-based approaches, a detailed presentation regarding simulation basics for production systems is stated in this section. Lastly, in the end the impact of all mentioned theoretical topics is summarised.

2.1 Principles of value-added Process from a Logistics Point of View

Production is a value-adding process that transforms simple or complex input goods into value-added output goods. Tempelmeier describes the overall long-term goal of a company as "to make money". This target is especially in a production environment influenced by the factors **time, quality, profitability** and **flexibility**. The faster a process can be finished, the more value-adding activities can be achieved by a resource. The effort in trying to overcome the process **time** span quickly results in the common goal of short lead times including short idle, waiting and setup time.[8]

The performance of a production system can be measured regarding quantities, value and **quality**. Especially technical challenging products are strongly affected by quality and the resulting customer satisfaction. Production quality is expressed through low rejection rate or scrap amount, but moreover in functionality, reliability, durability and environmental sustainability of the finished product.[9]

From an economic perspective there are two possible principles regarding **profitability**. The maximum principle wants to achieve a maximum of production result by a given volume of resources. The minimum principle seeks to fulfil a pre-defined volume of production or revenue with as little effort or input-value as possible. As a result of globalisation of markets and the dynamic development of technology, there exists big pressure regarding costs on companies. Every market participant has to manage the following possible cost potentials individually: Economies of Arbitrage (using price advantages), Economies of Scale (using quantity advantages), Economies of Scope (using variety advantages), Economies of Speed (using time and speed advantages) and Economies of Structure (using advantages of changes). The resulting focus regarding resources and production factors has a high impact on logistics. From a logistics perspective the parameter profitability can also be titled as **cost**, with big impact

[8] cf. Günther, H.-O.; Tempelmeier, H. (2012), p. 3.
[9] cf. Zsifkovits, H. E. (2012), p. 112; Günther, H.-O.; Tempelmeier, H. (2012), p. 3.

on inventory, capacity utilisation and human resources. Additionally costs considerably affects the other three variables time, quality and flexibility.[10]

The last influencing factor is **flexibility**. The faster and better a production system can be adjusted regarding adaptation scope including positive resulting economic effects, the more flexible such an environment is. A similar term synonymously used is agility with the objective to focus on maximising the use of organisational resources and capital.[11]

All four performance objectives depend on each other and have to be individually balanced and defined for every considered system. In contrary to the mentioned objectives, Gudehus defines the key objectives of supply chain management represented by minimal costs, market-conform delivery time and high delivery reliability.[12]

Differentiation between production planning, scheduling and control

As value-adding processes of manufacturing companies are always about production procedures, planning and sequencing are highly relevant disciplines.

Focussing on simulation of the material flow within manufacturing process, three terms should be defined at the beginning of this thesis that are involved in setting up an optimised system:[13]

- Planning generally describes the selection, structure, dimension and optimisation of processes, networks and resources in order to fulfil **future performance requirements**.
- Scheduling or disposition is about the quantitative allocation of **current performance requirements (=orders)** including a timed assignment to available resources.
- Control directs the operational processes of a production or performance environment and **regulates the execution** of defined orders regarding amount, content and time.

Scheduling takes place in short time intervals, ranging from minutes to hours for express orders to days for cyclic scheduling processes. If it is done regularly in short cycles or executed in case of an express or large order, Gudehus calls this process *dynamic scheduling*. That logic ensures quick reaction to current happenings and adaptation of scheduling parameters. Doing so, this approach aims to assure delivery time in line with the market, cost-optimised delivery reliability and prevents overstocking as well as understocking. Based on customer orders, previous planning activities, demand forecasts, current inventory and available resources, a quantity of products to be manufactured results. Those scheduled orders should then be started and executed from production control, after ensuring that all needed resources like materials, equipment and operators are available. The primary goal of production control is to realise the

[10] cf. Günther, H.-O.; Tempelmeier, H. (2012), p. 3 p.; Pfohl, H.-C. (2018), p. 51; Zsifkovits, H. E. (2012), p. 12.
[11] cf. Günther, H.-O.; Tempelmeier, H. (2012), p. 4; Matt, D. et al. (2021), p. 199 p.
[12] cf. Bauer, J. (2017), p. 7; Gudehus, T. (2011), p. 8.
[13] cf. Gudehus, T. (2011), p. 3 pp.

planned finish dates in the best way possible, even in the event of unavoidable disruptions or deviations of the production process.[14]

A discipline increasingly spreading since the last 30 years is **Lean Production** which is basically not a technical achievement, but rather a paradigm shift. The principle is to align all production processes to the customer and simultaneously avoid wastes. Taiichi Ohno, as production system founder of Toyota, described their intention as, "All we are doing is looking at the timeline from the moment a customer gives us an order to the point when we collect the cash. And we are reducing that timeline by removing the non-value adding wastes."[15]

2.2 Logistics Key Performance Indicators

In order to overview and assess the performance of a system, it is essential to have appropriate key performance indicators (KPIs) in place that reflect and summarise the reality. This section focuses on measures that reflect the behaviour of an internal production system and discusses the relation between inventory, lead time, capacity utilisation, profitability and flexibility.

Lödding and Wiendahl define a couple of logistics KPIs and distinguish between external (customer oriented) versus internal logistics target values. Additionally, they separate between logistics performance versus cost indicators, an overview is shown in Table 1. The complete set of measures including a detailed description is mentioned in Lödding, H. (2016), which refers to the Hanoverian Supply Chain Model, published by the IFA (Institut für Fabrikanlagen und Logistik).[16]
Concentrating on internal production logistics, this thesis will especially outline lead time, due date deviation and reliability, inventory costs and capacity utilisation to determine the logistics performance of a job-shop production environment.
Default costs and external measures won't be further described because this thesis focuses on internal processes where no costs for delays are charged, nor external impacts are taken into account.

	logistics performance	**logistics costs**
external	*make-to-order* delivery time delivery date deviation delivery reliability	price
	make-to-stock service level	
internal	lead time due date deviation due date reliability	inventory utilisation default costs

Table 1: Logistics KPIs[17]

[14] cf. Wiendahl, H.-P. (2010), p. 258; Gudehus, T. (2011), p. 4 pp.
[15] cf. Reinhart, G. (2017), p. 33 p.; Ōno, T. (2013).
[16] Nyhuis, P. (2017).
[17] Lödding, H. (2016), p. 22.

Due date reliability describes the percentage of handled orders within the planned timespan, in comparison **due date deviation** reflects the delta between actual and planned due date. Lödding defines **lead time** (*German 'Durchlaufzeit'*) as time duration between an order is released and finished in the shopfloor, which is often simplified reported in days. In make-to-order (MTO) production it represents the lower limit of the total order delivery time, and its variance further influences due date reliability and due date deviation. Regarding cost indicators, especially amount of inventory and **capacity utilisation or occupancy** are relevant KPIs related to the state of a shopfloor. The difference between utilisation and occupancy rate will be defined below Figure 1. Especially utilisation has a strong relation to the physical value performance, which is defined as the ratio between work and time. In correspondence to logistics processes, performance is often described as output or throughput within hours, days or weeks. The term **capacity utilisation** is further defined as the proportion of average and maximum possible performance, regarding the available capacity of a work system.[18]

As last undefined internal metric of Table 1, **inventory** can be split into warehouse and production inventory. Firstly, talking about raw materials, semi- and finished goods stored in a defined storage location. Secondly, materials related to production orders which are released for processing but not finished yet, are called **WIP (work in process)**. Usually, these materials are physically located in the manufacturing area.[19]

The purpose of inventory management is an integrated view regarding the complete stock of a company including the goal to increase profitability through an increased capital rotation rate by reducing fixed capital caused by inventory. Nevertheless, warehouses and storage buffers are essential connection elements of physical goods in production systems to ensure a fluent material flow, synchronising supply and demand. Furthermore, material reliability is a pre-condition to even start a production process, thus interim storage is necessary - only the determination of the right size is crucial.[20]

The more inventory a company has, the more money is locked in its current assets and can't be used for different investments or payments. Additionally, interest costs have to be taken into account which increase in proportion to the capital tied up if the interest rate remains constant. As soon as inventories go down, interest costs are reduced and profit increases accordingly. Besides the mentioned financial reasons less inventory simply needs less space and results in a neater and easier controllable company. From a logistics perspective especially WIP influences capacity utilisation and lead time of orders, both KPIs rise with higher WIP. This further impacts the general output performance and due date reliability.[21]

This conflict of objectives is known for decades, Gutenberg called it the dilemma of scheduling (*German 'Dilemma der Ablaufplanung'*) which is especially dominant in a job-shop production environment: The ideal lead time would result if transition times between processes were zero or at least minimised, which can be achieved if buffers between them are scaled down as well. This objective of reducing lead times is countered by a

---

[18] cf. Lödding, H. (2016), p. 21 pp.; Nyhuis, P.; Wiendahl, H.-P. (2012), p. 26 p.
[19] cf. VDMA 66412-1:2009-10, p. 6; Lödding, H. (2016), p. 36.
[20] cf. Pfohl, H.-C. (2018), p. 52; Bauer, J. (2017), p. 162.;
[21] cf. Lödding, H. (2016), p. 36 p.

second objective which tries to utilise operating resources in such a way that the most favourable possible utilisation is achieved by avoiding idle or down time. Trying to optimise both objectives result in a clear conflict. Ideal planning can help to find an optimum proportion of setting parameters, but a certain dilemma will remain. As example, a workplace or resource could be claimed by a couple of production orders at the same time or even for different lasting processing times which implicates possible congestions, so-called **bottlenecks**, in the material flow. Thus, inventory buffers backup processes prone to failure, as well as unlevelled capacities or other process weaknesses. In a continuous or automated line production the dilemma can be levelled better, which leads to higher output performance. This is one reason why processing lines have emerged that much in the last 100 years.[22]

Another notable value associated to lead time of a production order is the **execution time** (*German 'Durchführungszeit'*) only containing setup and processing time which results by subtracting **transitional times** like transport or waiting time before and after processing. Different occurring times in production are shown in Figure 1. A resulting KPI by dividing lead time through execution time is the so-called **flow rate** (*German 'Flussfaktor'*), which can be 1 at its best, but results probably bigger. Nyhuis and Wiendahl state, that this metric is proportional to the amount of WIP. Differences can only be root caused to varying utilisation rates.[23]

Literature often mentions **Little's Law** when setting the mean lead time (LT_m) through a system in context with the mean number of orders (N_m) divided by the mean arrival rate (λ), represented by formula 2.1, which is a common law in queuing theory. This rule can be transformed by setting the flow rate (FR_m) in context with WIP (WIP_m) and mean workplace utilisation (U_m), shown in formula 2.2:[24]

- $LT_m = \frac{N_m}{\lambda}$ (2.1)
- $FR_m = \frac{100*WIP_m}{U_m}$ (2.2)

The detailed formula transformation can be studied in Lödding, H. (2016). Summarising it shows the connection between the relevance of waiting time in queues in front of workplaces (regarding WIP), the utilisation of a workplace assuming constant availability (considering performance) and the flowrate (reflecting lead time). This perception is used to define WIP, lead time and workplace utilisation as focused metrics for this thesis.

Lödding describes the KPI capacity utilisation as traditionally dominant, especially focussing on a high utilisation of expensive machines in order to ensure refunding. From an economical view, the truth is that after a completed investment, these so-called sunk costs shouldn't influence short-term decisions of production planning. They should not be decision relevant anymore because machine size and resulting price should have rather been calculated correctly in the past. That's why a directive to utilise expensive equipment on purpose on a high level is not economically justified and leads in many

[22] cf. Gutenberg, E. (1969), p. 213 p.
[23] cf. Nyhuis, P.; Wiendahl, H.-P. (2012), p. 21 pp.; Lödding, H. (2016), p. 59 p.
[24] cf. Lödding, H. (2016), p. 32 pp.

cases to higher inventory and longer lead times, whose effect has already been discussed.[25]

As last not mentioned influence factor, flexibility experiences a continuous rise of importance in recent years due to dynamic markets and shorter resulting response times. The necessity of flexible organisational structure on the one hand, and flexible production and logistics system on the other hand become obvious if unlevelled market demands or shifting capacities because of changing resource and material availability are investigated. Organisational flexibility enables faster and better decision processes in unforeseen situations and a flexible production and logistics system can be achieved by moving customer specific details closer to the final process steps. This can be realised in designing products in modular ways, establishing small production lots, reducing setup costs and arranging universally applicable operating equipment and employees.[26]

Having discussed the relations of relevant KPIs, the VDMA norm 66412-1 provides a standard to split the total available production time into the shown subsets in Figure 1.

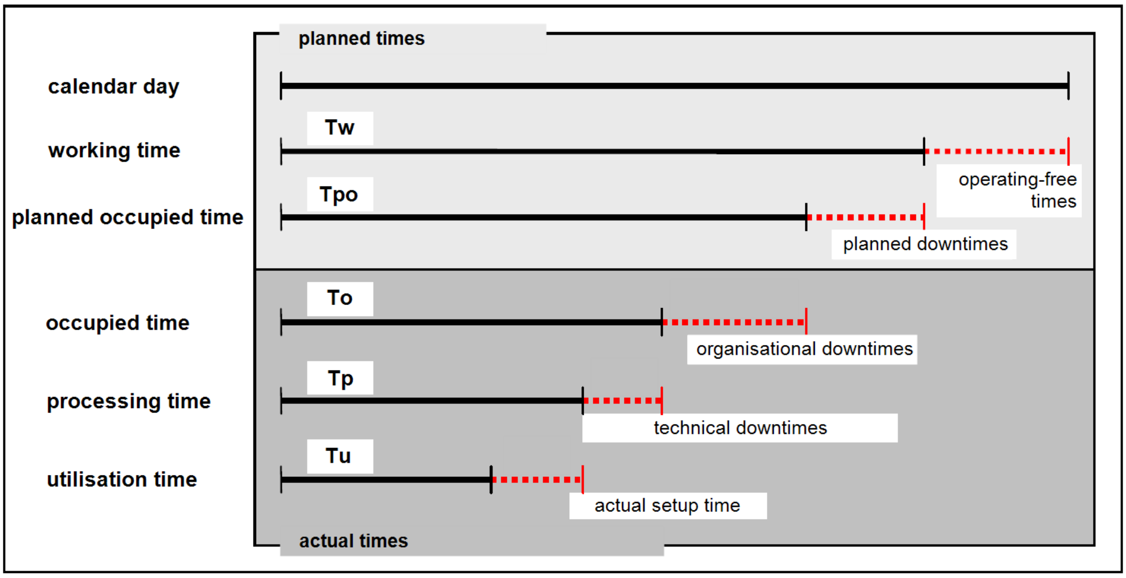

Figure 1: relation diagram for production time units[27]

According to Figure 1, a manufacturing calendar day consists of:[28]

- The total shift time represents the maximum available time for production.
- The working time T_W (*German 'Betriebszeit')* represents the planned time an operator or resource is available at the workplace for production or maintenance and covers shift time reduced by breaks and allowance time.
- The planned occupied time T_{PO} (*German 'Planbelegungszeit')* **excludes planned downtimes**, such as scheduled service or routine maintenance jobs. This resulting time period represents available time for production scheduling to allocate released production orders.
- The occupied time T_O (*German 'Belegungszeit')* describes the actual time a workplace is scheduled with an operation. Due to operational circumstances, a delta results compared to T_{PO}. This is because organisational downtimes such as

[25] cf. Lödding, H. (2016), p. 40.
[26] cf. Pfohl, H.-C. (2018), p. 53.
[27] VDMA 66412-1:2009-10, p. 9.
[28] cf. VDMA 66412-1:2009-10, p. 8 p.

handling errors, missing tools, missing energy or missing materials happen. Other reasons involve 'wastes' like transport or laytime that results as idle time of a workplace. This duration is also called execution time, meaning how long it has taken to finish an operation at a considered workplace, including machine breakdowns that have occurred.

- The processing time T_P (*German 'Bearbeitungszeit')* further excludes technical downtimes due to machine malfunctions or tool defects.
- Finally, the utilisation time T_U (*German 'Hauptnutzungszeit')* remains. This duration only includes the value-adding processing time an order is processed at a workplace, subtracting the optional necessary setup time T_S (*German 'Rüstzeit')*.

Having those duration differences defined, a couple of KPIs can be derived:[29]

- The utilisation degree (*German 'Nutzgrad')*, also called productivity, is a factor comparing the processing time T_P against the total occupied time T_O and gives information about the productivity of a workplace. Since only processing time is adding value to a product which is remunerated by the market, it is the goal of a company to optimise this metric.
  - $utilisation\ degree = \frac{T_P}{T_O} * 100\%$ (2.3)
  - In comparison, the VDI standard 3423:2011 calls this metric in *German* '*Nutzungsgrad'* and includes the duration for planned maintenance to T_O. Thus, the VDI KPI would result in a lower and stricter KPI.
  - A similar KPI focussing on operators rather than on workplaces is the operator productivity (*German 'Mitarbeiterproduktivität')*, which compares the order-related working time vs. the total paid presence time.
- The metric availability (*German 'Verfügbarkeit')* results, if the duration T_P is compared to T_{PO}, which includes operational downtimes. This KPI gives information about how much available capacity has been utilised for value-adding purposes.
  - $availability = \frac{T_P}{T_{PO}} * 100\%$ (2.4)
- The occupancy rate (*German 'Belegnutzgrad'*), sometimes also titled as capacity utilisation rate, is the ratio of time a workplace has been occupied by an order (T_O) versus the time a workplace should have been occupied (T_{PO}). This measure gives information about how much capacity of a workplace has been utilised and how much capacity has not been used (=idle time).
  - $occupancy\ rate = \frac{T_O}{T_{PO}} * 100\%$ (2.5)
- The process or technical availability (*German 'technische Verfügbarkeit'* or *'technischer Nutzgrad')* refers to the efficiency of a machine or workplace. It is the ratio between the processing time T_P and the occupied time T_O, excluding setup time. This metric indicates how much capacity could be increased if breakdown times were reduced.
  - $process\ availability = \frac{T_P}{T_P + breakdown\ time} * 100\%$ (2.6)
- The setup rate (*German 'Rüstgrad')* is an index which represents the percentage of setup time versus the total handling time T_H. The higher the value, the more

[29] cf. VDMA 66412-1:2009-10, p. 9 pp.; VDI 3423:2011-08, p. 8.

time is used for setting up workplaces instead of actually produce and adding value the products.

- $setup\ rate = \frac{setup\ time}{T_H} * 100\%$ (2.7)

Further KPIs regarding workplaces involve the number of items waiting in front of it, as also the resulting waiting time. If a processing sequencing is defined, commonly the term **queue** is used. The so-called **dwell time** includes setup and processing time to the waiting time.[30]

2.3 Bottleneck Identification and Evaluation

As mentioned in the previous chapter, bottlenecks in a production system arise because of congestions in the material flow. Since it is a common dilemma with big influence regarding productivity, many employees and researchers have tried to eliminate bottlenecks out of various systems. Thus, multiple authors have defined the term "bottleneck" in the past: Krajewski (2009) describes a bottleneck as a function limiting output, Roser and Nakano (2015) explain it as a process which influences a whole system by slowing it down. The bigger the influence, the greater the bottleneck. Chase and Aquilano (1992) define it as a resource whose capacity is lower than the demand.[31]

Zsifkovits explains a bottleneck as a machine, function, department or resource that has the highest occupancy rate in a considered period, which implicates a limitation of flow through a whole system.[32]

In general bottlenecks can be distinguished in static ones on the one hand, which are unchangeable regarding their condition and exist most of the time caused by design causes. On the other hand, there exist dynamic bottlenecks which are time-variable and changing their position depending on the current situation. Such shifting bottlenecks result from variable process parameters like different work content (product mix), breakdowns or changing market demand (product volume and mix). Especially in dynamic systems bottlenecks tend to shift, which further implies that different processes might result as a limiting resource. Additionally, bottlenecks can emerge from outside a company's sphere of influence, such as material availability, transport limitations or interruptions and political riots. A common visualisation method to explain the main characteristics of a bottleneck is the funnel model, which is shown in Figure 2 a).[33]

---

[30] cf. Hedtstück, U. (2013), p. 80 p.
[31] cf. Roser, C. et al. (2015), p. 2.
[32] cf. Zsifkovits, H. E. (2012), p. 109.
[33] cf. Klenner, F. et al. (2016), p. 541; Roser, C. et al. (2015), p. 2.

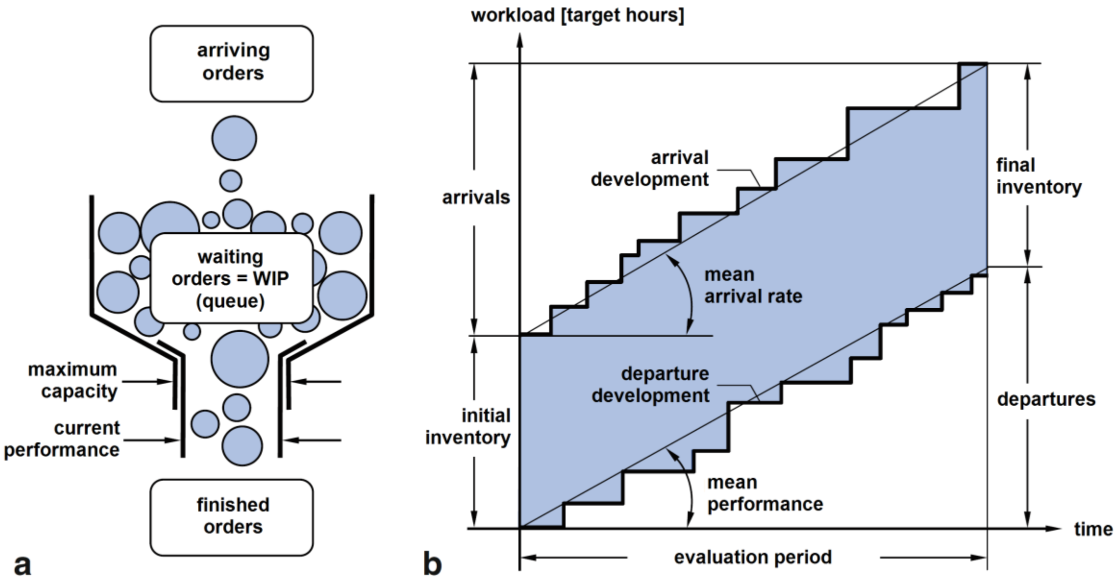

Figure 2: a) funnel and b) throughput model[34]

The shown throughput model in Figure 2 states the development of inventory in general, which can either be WIP in front of a workplace, stock in a warehouse or customers in front of a bank clerk. Every resource unit can be described with the throughput parameters arrivals, inventory and departures. Arriving elements like orders, materials or customers are added to the already waiting elements in front of a resource and get processed or served accordingly to the current performance. The throughput model demonstrates the accumulated workload of processed orders regarding the performed completion time as departure development. Similarly, the arrival development results from arriving workload of orders along the evaluation period. The start of the arrival line is defined by the level of initial inventory in front of a resource. In the end, the amount of unprocessed orders represents the level of final inventory. The mean gradient of the arrival development is called arrival rate, equivalently the mean gradient of the departure development is defined as mean performance. In order to avoid a constantly increasing queue in front of a resource, arrival and departure rate (defined as mean performance), must run parallel on a long term, considering a stable and stationary system.[35]

The weakest link determines the strength of a chain. This well-known statement also leads to the famous five step process by Goldratt and Cox presented in the book 'The Goal' which underlines the basis for the concept Theory of Constraints (TOC), defining an approach to eliminate bottlenecks of an analysed system. This methodology, shown in Figure 3, firstly focusses on increasing overall productivity of a considered system by excluding the dominating constraint(s) and secondly reducing inventory which is limiting throughput. Those five steps implicate:[36]

1. Identify the system's constraint(s).

[34] source: Bechte (1984) cf. Nyhuis, P.; Wiendahl, H.-P. (2012), p. 25.
[35] cf. Nyhuis, P.; Wiendahl, H.-P. (2012), p. 25 p.; Lödding, H. (2016), p. 60 pp.; Gudehus, T. (2011), p. 37.
[36] cf. Goldratt, E. M.; Cox, J. (1993), p. 296 pp.

2. Decide how to exploit the system's constraint(s): make sure that the identified constraints are fully utilised by ensuring that operators are present when needed for example.
3. Subordinate everything else to the above decision: ensure that all other processes of the system support the needs of the constraint(s), by ensuring its/their utilisation for example.
4. Elevate the system's constraint(s): if the bottleneck is still exists, take further actions in order to eliminate the identified constraint(s), this step may be linked to capital investments as adding an additional resource or changing the material flow for example.
5. If in the previous steps a constraint is broken which means that the system's constraint has changed, repeat the whole process and don't allow inertia to get a constraint.

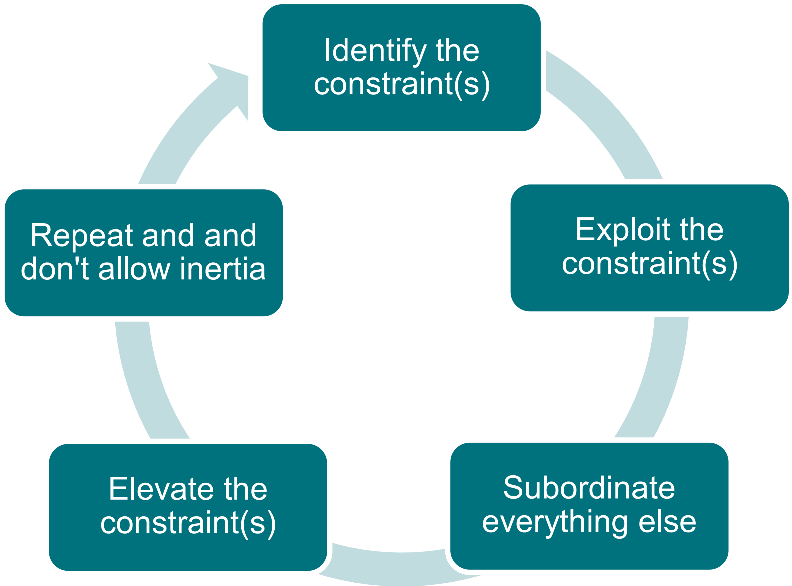

Figure 3: five focusing steps of TOC

A relevant factor mentioned in Goldratt's 'The Goal' is to additionally consider the customer or market as a system's bottleneck. Identifying this and other limiting factors that are not directly associated with workplace performance, lead to the term 'constraint' instead of 'bottleneck'. Such additional influence factors may be logistics supply processes or a critical process within the information flow.[37]

Based on TOC, Goldratt proposed a production control system (PCS) called drum-buffer-rope (DBR), where the identified bottleneck of a system takes the central role as drum, visualised in Figure 4. This PCS is especially designed for MTO production systems, ensuring that the identified bottleneck defines as drum the schedule of the total system and is never starved of material, thus a buffer is set in front of it. The term 'rope' in the method's name represents a signal or information for the system's release process, connecting the buffer level to the first workplace. As soon as the drum processes a part and takes material out of the buffer in front of it, additional orders are allowed to enter the system's queue. The optional space buffer after the drum signifies the importance

[37] cf. Roser, C.; Shook, J. (2021), p. 325 pp.; Goldratt, E. M.; Cox, J. (1993), p. 297; Roser, C. et al. (2015), p. 3;

that the bottleneck should not be allowed to be blocked by its successor in means of always being able to push material forward. This PCS has similarities with another PCS called CONWIP (constant work-in-process), which will not be further explained, but reference is made to the cited sources Roser, C.; Shook, J. (2021) and Gómez Paredes, F. J. et al. (2022).[38]

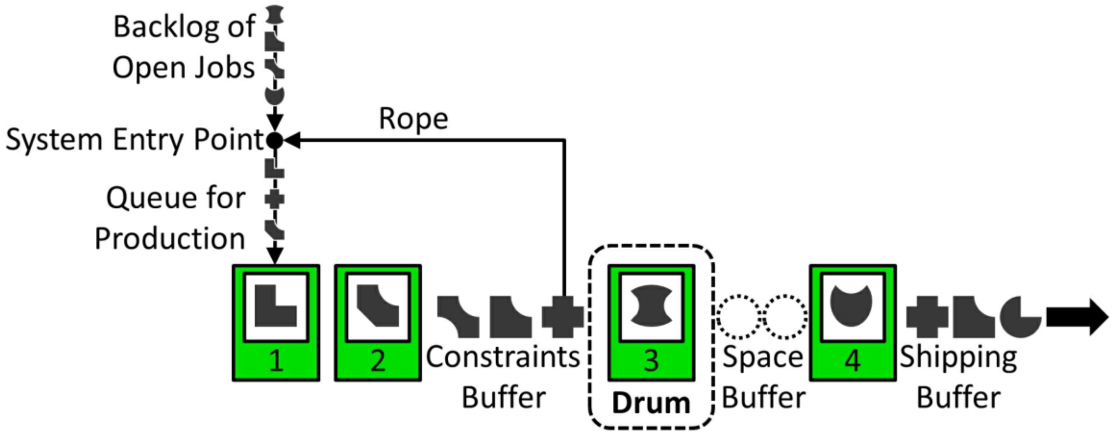

Figure 4: example of a DBR system[39]

Bottleneck identification methods

In order to follow the five steps mentioned in Figure 3, it is essential to determine the current constraint(s) of a considered system. Nowadays, there exist a couple of common bottleneck detection methods in the industry such as **Process Time, Utilisation or OEE-based approaches**. These methods consider the process time of the material flow under isolated conditions as input and calculates resulting bottlenecks on a mean basis. The application is simple and fast, but it has the disadvantage that only static bottlenecks can be identified because no dynamic system properties are taken into account. Parameters that cause a gap between net production and total available time like losses, wastes or breakdown which result as reduced available capacity can only be covered on as averaged values. One example for this method is value stream mapping (VSM) which can determine the maximum system capacity under ideal or mean conditions.[40]

Simulation is a technique that enables an experimental procedure to represent a system including its dynamic processes with the help of a computerised model. Defined parameters that exist in the real environment can be adapted and optimised in order to achieve a previously specified model target. A system can also be modelled, if it does not exist yet. As disadvantage the challenge to collect correct statistical data can be mentioned, which is needed to take appropriate parameter distributions into consideration. Adding a simulation extension on top of a VSM study effectively resolves

[38] cf. Gómez Paredes, F. J. et al. (2022), p. 643; Roser, C.; Shook, J. (2021), p. 325 pp.
[39] Roser, C.; Shook, J. (2021), p. 326.
[40] cf. Shou, W. et al. (2021), p. 2131 p.; Roser, C. et al. (2015), p. 3 p.

the limitation of only considering a static view. Among all simulation techniques, discrete-event simulation is the most commonly used approach as VSM enhancement.[41]

The **Active Period Method** is also able to identify shifting bottlenecks in line and job-shop environments by clustering processes into active and passive ones. Active processes are defined as not blocked or waiting which include the states of workplaces such as production, setup or downtime. Blocked or passive states refer to situations where a workplace is impeding for its successor in order to push material forward or waiting for materials of the predecessor to continue processing. This method is based on the assumption that the longer the active period of a process is, the more likely it is that this process will limit the output of others. Consequently, the process with the longest active period results as the system's bottleneck. Important to note is, that an active period includes all active states that are not interrupted by a passive state. This approach requires like simulation a high data quality to know about the durations of active and passive states of each workplace. This technique definition implicates, that the active period method is very similar to measuring the occupancy rate of workplaces, as already defined KPI in chapter 2.2.[42]

The **Real-Time Bottleneck Detection including Prediction method** focusses on buffers between production lines and assumes that the bottleneck is detectable by looking at the buffer size in front of each workplace. If first several connected processes have buffers filled to the maximum and afterwards certain workplaces have empty ones, the bottleneck is located before the first empty buffer. If no clearly empty or full buffer exists, this method focusses on arrival and departure rates of each buffer. The workplace which is associated to the buffer having the lowest departure rate will be identified as bottleneck. An additional advantage is, that the duration until the failure of one process affects the production of another bottleneck is determinable, thus this method enables a predictive bottleneck detection based on process failure. Summarising, this method focusses on queue waiting times and queue or buffer departure rates in front of workplaces.[43]

The **Bottleneck Walk** evaluates bottlenecks of flow lines by performing a shopfloor walk and noting the inventory levels of resulting bottlenecks during the observation. There exist certain rules that define which kind of notes should be taken accordingly to particular buffer levels to derive correct statements afterwards. As this method is designed for production lines it will not be explained in further detail, for more information the source Roser, C. et al. (2015) delivers adequate input.[44]

The **Smart Data Model** enhances the bottleneck walk, which requires manually documented buffer levels in front and after evaluated processes, by determining buffer levels automatically. This is made possible by detecting material transports based on confirmed bookings documented in the production control system. Those bookings act as input for this approach and enable the virtual representation of current buffer levels. According to the resulting amount of inventory, again certain rules are set to identify

---

[41] cf. Huynh, B. H. et al. (2020), p. 16; Roser, C. et al. (2015), p. 3 p.; Shou, W. et al. (2021), p. 2131 p.
[42] cf. Klenner, F. et al. (2016), p. 543; Roser, C. et al. (2015), p. 4.
[43] cf. Wedel, M. et al. (2015), p. 142 pp.; Klenner, F. et al. (2016), p. 543 p.
[44] cf. Roser, C. et al. (2015), p. 4 p.; Klenner, F. et al. (2016), p. 544.

bottlenecks regarding the current buffer levels. For more information the source Klenner, F. et al. (2016) includes detailed instructions. As pre-condition high data quality has to be available regarding material transports, otherwise this technique is not applicable.[45]

This list summarises used approaches in industry to identify bottlenecks in various systems. Following Goldratt's TOC the second step is to exploit a bottleneck and then subordinate all other processes to enable the maximum utilisation of the identified bottleneck. Possible actions include shifting workload to other resources if possible, plan additional shifts for the bottleneck workplace or add additional resources. Further procedures include the reduction of downtimes to increase availability or reduce transport and idle times to increase the occupancy rate as also plan to carry out maintenance jobs at non-critical times.

2.4 Simulation of Production Systems

As described in the previous chapter, simulation is one possibility to determine bottlenecks in dynamic systems based on available capacity, current level of inventory, and planned orders to be processed. The core intention of this thesis is to evaluate the simulation approach in a job-shop production environment including the consideration of dynamic system behaviour. Thus, the development of a discrete-event simulation model is a pre-condition to evaluate scenarios of different production volumes based on a validated model.[46]

To ensure a common understanding of all performed steps and resulting conclusions, this chapter gives insight about the theoretics of simulation methods and their approach, focusing on production systems. The first pages start with defining some basic terms and characterising model differences, followed by a more extensive reflection of DES. Moreover, the verification and validation (V&V) procedure model developed by Rabe, Spiekermann and Wenzel is described, including some V&V techniques in the last subchapter.

The benefits and potentials of simulation and especially the application of DES have been proven over the years in various fields ranging from production systems, transportation, warehousing, computer systems, health, defence and business process management. Particularly initiatives in developing a digital factory have shown that significant improvements in planning and decision making can be achieved. The term digital factory sums up the interaction of models, methods and tools in form of simulation and three-dimensional visualisation integrated into consistent data management, having the goal for holistic planning, evaluation and ongoing improvement of all essential processes and resources. For that reason, the necessity of using simulation for planning and operational management of logistics systems is no longer questioned.[47]

For a common basic understanding, **models** are always abstracted representations of an examined system with defined boundaries and limitations including a purpose or set

[45] cf. Klenner, F. et al. (2016), p. 544 pp.
[46] cf. Lindegren, M. L. et al. (2022), p. 30 p.; Murphy, A. et al. (2020), p. 1.
[47] cf. Brailsford, S. C. (2014), p. 10; Tempelmeier, H. (2018), p. 1; Bracht, U. et al. (2018), p. 11.

of questions they are capable to answer. Models can help analysing and understanding system behaviours and interactions between components and are especially useful if interdependencies within the considered environment get too complex for the human mind. Abstraction and idealisation are used to virtually transform and represent the real world, thus a model always results as a simplification of the reality. Deciding which system properties are essential and which can be ignored depend on the model's purpose. During the development process of a model, V&V has to ensure that all relevant properties are covered.[48]

Simulation is a term with a big number of possible definitions stated in literature. The common understanding is that simulation is the process about reproducing dynamic processes of a system in an experimental model with real data and striving for undiscovered insights which are replicable for the reality. Additionally, predictions about the real system are tried to be retrieved by analysing the trace of model's states over the simulation time from some initial state. Digital simulations are realised by coded software programmes and single simulation experiments are called 'runs', executing a specific data set or variable settings.[49]

Simulation has become an important tool in order to assist decision makers in wide variety including production and logistics purposes. Nevertheless, one of most important criteria is the validity of a simulation model, that represents the 'reality' at least regarding the design goals sufficiently and offers useable results.[50]

How to achieve such results will be discussed in this entire chapter.

2.4.1 Definition of Simulation Methods and their Components

First of all, different simulation method classifications and their components will be discussed in this subchapter.

Systems can be described as discrete or continuous, only a few of them are able to be categorised as completely discrete or continuous, but usually one type of change dominates. In a **discrete system**, a variable can change its state only at a discrete set of points in time, such as the number of customers in a queue in front of a counter, shown in Figure 5. Another example would be the processing lifecycle of an order including arrival, process, finish and departure at discrete timestamps, illustrated in Figure 8 in chapter 2.4.2. In contrary, a **continuous system** describes the continuous change of a state variable over time. As example, Banks (2013) mentioned the water level behind a dam, which rises during/after a rainy day and decreases afterwards again, visualized in Figure 4. Such systems are described by a set of coupled differential equations.[51]

[48] cf. Page, B.; Kreutzer, W. (2005), p. 5 p.
[49] cf. Hedtstück, U. (2013), p. 3; Page, B.; Kreutzer, W. (2005), p. 9; Rabe, M. et al. (2008b), p. 12.
[50] cf. Page, B.; Kreutzer, W. (2005), p. 195 p.; Sargent, R. (2010), 168–169.
[51] cf. Banks, J. et al. (2013), p. 12; Tempelmeier, H. (2018), p. 10.

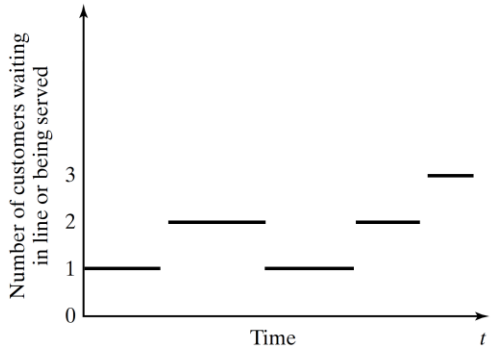

Figure 5: discrete-system state variable[52]

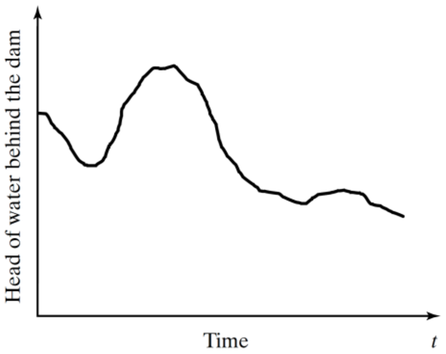

Figure 6: continuous-system state variable[53]

Continuous systems are defined by changing constantly over time and focus on physical-technical applications, such as the simulation of air flows on airfoils. Common techniques are finite element method (FEM), multi-body or kinematics simulation describing for example forces resulting from springs, dampers and actuators. A special variant regarding kinematics is about modelling handling processes of humans and analysing the impact on ergonomics, known as ergonomics simulation.[54]

Prior to an explanation of components and terms used in the field of simulation models, briefly other simulation types or also called **simulation paradigms** besides DES are mentioned.

System Dynamics (SD) models are categorised as continuous and represent the average flow through a system instead of focussing on individual events. That enables a more general and macroscopic view on a system instead of single measures which could lead to inappropriate conclusions. It is characterised by the ability to represent feedback and delays, exploring which impact a system's structure has on system behaviour. A good example is mentioned in Einzinger (2014), where inventory levels give feedback as soon as they are below a certain threshold. The levels will be changed after the placed order is delivered, which happens after a delay of transport and handling effort.[55]

Another mainly continuous simulation type is called **Dynamic Systems (DS)**, modelling physical state variables at a very low abstraction level. It is used in electrical, mechanical, chemical and other technical engineering disciplines, using block diagrams as typical graphical modelling language to describe a dynamic system like a spring for example. The model is described by a number of state variables including algebraic differential equations. In contrast to a SD model, DS variables have a defined 'physical' meaning, like acceleration, velocity, location or pressure, for one entity without any aggregation. MATLAB Simulink is handling such models for example, also capable to handle SD problems, but not supporting the way SD modelers are used to perform simulations.[56]

---

52 Banks, J. et al. (2013), p. 12.
53 Banks, J. et al. (2013), p. 12.
54 cf. Bracht, U. et al. (2018), p. 118 pp.
55 cf. Morgan, J. S. et al. (2017), p. 908; Einzinger, P. (2014), p. 27 pp.
56 cf. Borshchev, A.; Filippov, A. (2004), p. 3 pp.

The newest of all paradigms, is called **agent-based modelling (AB)**. It is mainly applied to discrete systems with a wide range of abstraction levels, varying from single robots, to clusters of customers and up to competing companies. Many different definitions exist about the discussion which properties an object must have in order to be called ‘agent’. Examples are the ability to learn, pro-active or spatial awareness. A notable feature of all AB models is the fact that they are essentially decentralised. An agent can be described as autonomous entity which is able to notice its environment, other agents and based on these settings able to make decisions. It is defined by attributes and follows certain rules which influence its behaviour. As an agent may have the capability to learn, its rule set will be adapted which will result in a change of its behaviour as well. Law (2014) defined AB as a variation of DES based on conducted expert interviews because system state changes occur at defined number of points in time.[57]

Before focussing on discrete systems and DES as fourth simulation paradigm, some basic terms and components used in the context of simulation are defined:[58]

- A **system** is the summarised collection of entities (like operators, workplaces or machines) that are interacting with each other or not, trying to achieve one or more goals. These entities represent components of the system.
- The **system state** is the compilation of all variables defining a system at any time. Examples are the remaining shift time, number of production orders in front of a workplace or the current level of inventory.
- An **entity** is an object or component which is part of the represented system and needs to be represented in the model. Entities can be dynamic by ‘moving’ through the system, such as a customer, or static like a bank clerk.
- **Attributes** define properties of entities which can either change over time or not. Examples might be the value of a material, the planned process time, the waiting time or the priority of an order in a queue.
- A **resource** is an entity which serves dynamic entities by providing capacity for activities, such as service or value-adding processes. A demanding entity can request units of a resource, if the resource denies the request the entity joins a queue or takes other actions (like requesting a different resource for example).
- An **event** is atomic and cannot be further decomposed, it does not consume simulation time. The execution of an event changes the system’s state and remains valid in the simulation model until the next event. An event can be created from outside of the model (exogenous) or from within the model due to a previous state or event (endogenous). A general example would be the arrival of a new customer or order.
- The **event list**, also called future event list (FEL), holds record about future events and is ordered by the time of occurrence. With every simulation step, the dependencies between all future events are re-calculated, maybe changed or new events added.

[57] cf. Law, A. M. (2013), p. 694; Borshchev, A.; Filippov, A. (2004), p. 6.
[58] cf. Banks, J. et al. (2013), p. 91 p.; Tempelmeier, H. (2018), p. 11 p.; Banks, J. (1998), p. 6 pp.; Page, B.; Kreutzer, W. (2005), p. 27 p.

- An **activity** represents a time-related operation, which transforms the state of one or more entities. The duration of an activity is latest known when it begins and can either be a constant (e.g. as an input from an external file), a random value as result of a statistical distribution, a result from an equation or calculated based on an event state. It is always characterised by a start and an end event.
- A **process** or **lifecycle** represents the planned events and activities of an entity.
- **Queues** or regularly just called **lists** are a collection of associated entities that are ordered in a defined sequence in front of a resource. The sequence logic can vary from FiFo (first in, first out) to LiFo (last in, first out) or more complex priority rules.
- The **simulated time** or **clock** is the basis for start- and endpoints of occurring events. Compared to real time, which is actually passing during a simulation run, the model time is fictitious and not related to the length of computation. It has a strong influence, since the system's state can only change to the least multiple of a chosen unit like milliseconds, seconds or minutes for example (only valid statement for discrete simulations).

Since this thesis focuses on discrete systems, the variants of that category are further discussed more deeply.

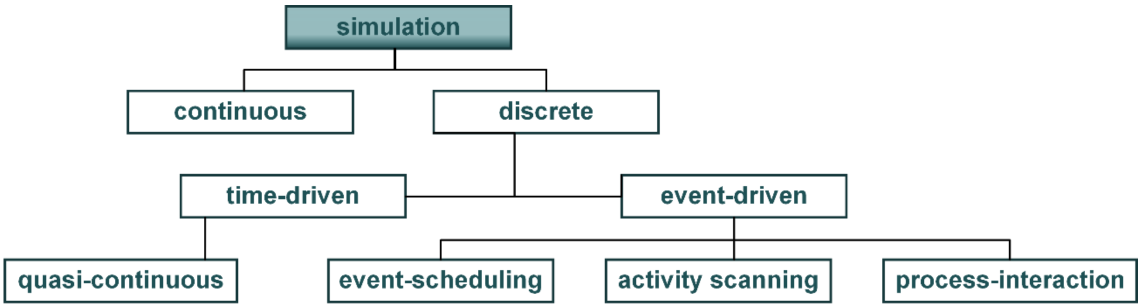

Figure 7: classification of simulation methods[59]

Mattern and Mehl already classified in 1989 the existing simulation methods shown in Figure 7, that execute changes of model time and states differently. Like mentioned before, a variable of a discrete system can only change its state at defined points in time. The **time-driven method** or so-called **fixed-increment time advance method** has a defined timestep Δt, which is added in every simulation step. Changes of states within one period of Δt are executed after the respective step. A small chosen timestep results in high computing time and close behaviour to a continuous model, thus sometimes also called as **quasi-continuous method**. As this approach is not considered as highly relevant for logistics purposes of production systems, it won't be discussed more detailed.[60]

More relevant methods of studied literature taken by the simulation community are described as **event-oriented** or **next-event time advance mechanism**. All following methods have in common, that a system's state changes based on happened events.

[59] cf. Mattern, F.; Mehl, H. (1989), p. 200; Tempelmeier, H. (2018), p. 12.
[60] cf. Tempelmeier, H. (2018), p. 10.

This method can be split into three different main approaches, which can all be applied in DES models and are briefly defined as follows:[61]

- **Event-scheduling approach**: A simulation analyst focuses on events and their resulting effect on the complete system and its state. This approach splits the considered system into a set of events that occur at specific times and cause a change of states. In order to apply this method, the future has to be predictable to sufficiently long intervals. E.g. an order is started at a workplace and will be finished after a certain period of time.
- **Activity-scanning approach**: This method focuses on the activities taking place in a system. In a cyclic time increment, conditions for every activity are checked to see which of them can be triggered. Applying this method, a model results consisting of independent modules that wait to be executed or activated ('triggered'). E.g. a transporter gets activated as soon as an order is ready for pick-up at a workplace.
  Disadvantages of this method are repeated scanning steps which can result in slow runtime on computers. Thus, this approach has been modified to the so-called **three-phase approach**, which keeps the main advantages of the original proceeding, but combines it with features of event-scheduling and allows the avoidance of scanning if it is not necessary.
- **Process-interaction approach**: This technique is based on a process view and organises the model into interactive parallel processes that change state variables or wait for each other. The model is defined by entities or objects and their respective lifecycle how they flow through the system. This lifecycle is characterised by a process of time-demanding activities and resulting delays because of requesting resources with limited capacities. The consequences are evolving queues and waiting times, which can be influenced by setting rules and priorities, such as FiFo (first in, first out), shortest processing time, earliest due date or minimum operation slack time (operation due date – remaining operation time – time now).
  As sub-method the **transactional flow method** additionally distinguishes between mobile dynamic objects ('transactions') and permanent stationary objects ('stations'), which is commonly used in queue systems.

It should be noted that some authors define the three-phase and transactional flow method as separate simulation approaches.

2.4.2 Discrete-Event Simulation (DES)

Having System Dynamics, Dynamic Systems and agent-based modelling shortly described in the previous chapter, the fourth (or third, depends if AB is considered as an own paradigm) type is DES.

Modelling a system by discrete-event simulation is about representing a system's logic of physical or non-physical entities, moving through connected components including defined processes or machining stations and competing about existing resources. Those moving entities (objects) can be represented by semi-finished goods in a workshop, cars in a road network, information of business processes or customers in front of bank or

[61] cf. Banks, J. (1998), p. 9 p.; Tempelmeier, H. (2018), p. 10 pp.; Banks, J. et al. (2013), p. 94 pp.

pharmacy clerks. Most processes have a strong **stochastic** character, which means that random events occur whose point of occurrence and characteristics are **not predictable**. For such cases, simulations are used to get a process insight by using appropriate distributions of possible events such as breakdowns, object arrivals, processing or finish times.[62]

If a model does not consider any probabilistic components, like random occasions or durations, it is called **deterministic**. As an example, a chemical reaction can be mentioned, where the output is always **determinable** and equal, as soon as input parameters and relationships are defined. Another case would be considering breakdowns of a machine in a model by always applying the mean-time-to-repair, calculated based on historical values instead of sampling out of a distribution.[63]

The dynamics of a considered system applying DES are represented by dynamic entities travelling through a system. Occurring events change states of model components or trigger activities. This simulation type is often used to represent a system on an operational level, modelling the effect of dependencies and discovering the effect of stochastic impacts. During an entity's lifecycle it may stay in queues, gets processed or occupies and releases resources.[64]

As basis for the correct timing of events, the simulation clock has historically two different approaches in advancing its time: the next-event time advance (NETA) and fixed-increment time advance (FITA) method. The first approach is used by all major simulation software and most modelers, thus this thesis will also use NETA as simulation clock method. This implicates, that the clock is initialised to 0 and all known future events are determined and saved into the future event list (FEL). The simulated time is then advanced until the first and closest future event, where the system's state is recalculated including the known occurrence times of future events. After that, the simulation clock is advanced again to the next closest future event, where a system and FEL update takes place again. This process is repeated until a predefined stop condition is met, like a specified simulation time or system state for example. Due to the fact that in DES models system changes only take place at occurring events, periods without events will be skipped. The FITA method would not overjump such inactivity periods which can result in higher computing times.[65]

The difference about those two simulation clock approaches was already shown in Figure 7 and distinguished between time-driven and event-driven methods.

In Figure 8 exemplary relationships within a DES are shown including periods without any event which will be skipped based on the NETA method. The lifecycle of fictitious order 1 is represented by an arrival at event 0 (e0), followed by processing as activity 1, which ends at e4. A similar lifecycle is additionally shown for a fictitious order 2, both starting at simulation time 4. e7 shows the case, that new events can be created in every simulation step, e.g. e7 could be created based on e6 when order 2 gets finished.

[62] cf. Hedtstück, U. (2013), p. 39 pp.
[63] cf. Law, A. M. (2013), p. 6.
[64] cf. Morgan, J. S. et al. (2017), p. 908; Borshchev, A.; Filippov, A. (2004), p. 6.
[65] cf. Law, A. M. (2013), p. 7 p.

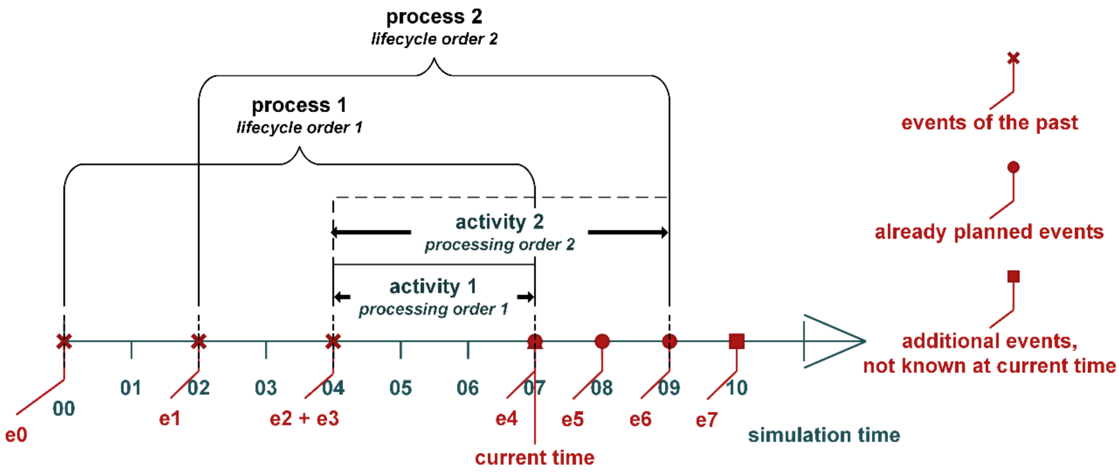

Figure 8: example for relationships between events, activities and processes[66]

2.4.3 Procedure Model of Simulation with Verification and Validation

It has already been discussed that a **model** represents an abstracted representation of the reality with a specific purpose. That's why a model can only be valid regarding the defined target. Box stated in 1987 "All models are wrong, the practical question is how wrong do they have to be to not be useful." As statistician and professional modeler, he was aware that there would probably always be some deviation between a model and the reality. But the central question is, which error-tolerance is allowed in order to conduct valid conclusions. Thus, his quote is not comparable with the known saying which is regularly associated with Winston Churchill: The only statistics you can trust are those you falsified yourself.[67]

To meet the goals of an expected model, verification and validation (V&V) criteria were established by Rabe, Spieckermann and Wenzel. Those include:[68]

- Completeness regarding content, structure and data.
- Consistency regarding semantic dependencies, content and structure.
- Accuracy regarding modelling, right choice about the level of detail, granularity of data and variable distributions.
- Currency regarding validity of content and time about used information and data including validity of a model for specific purpose.
- Applicability regarding fit accuracy, suitability and useability of model results, appropriateness of a model's purpose and the benefit for users.
- Plausibility regarding transparency of dependencies and model results.
- Clarity regarding reproducibility for users, transparency in modelling and reading.
- Feasibility regarding technical requirements, achievability of project targets and project timelines.
- Accessibility regarding data and document availability, credibility of information and data sources and procurement effort.

---

[66] own drawing, cf. Page, B.; Kreutzer, W. (2005), p. 28; Tempelmeier, H. (2018), p. 10; Mattern, F.; Mehl, H. (1989), p. 201.
[67] cf. Box, G. E. P.; Draper, N. R. (1987), p. 424; Georgiev, G. (05.11.2019).
[68] cf. Rabe, M. et al. (2008b), p. 22 p.

Based on the guideline of the German engineers' association VDI (VDI 2008), Rabe, Spieckermann and Wenzel designed a very often cited procedure model of simulations with V&V in production and logistics shown in Figure 9.

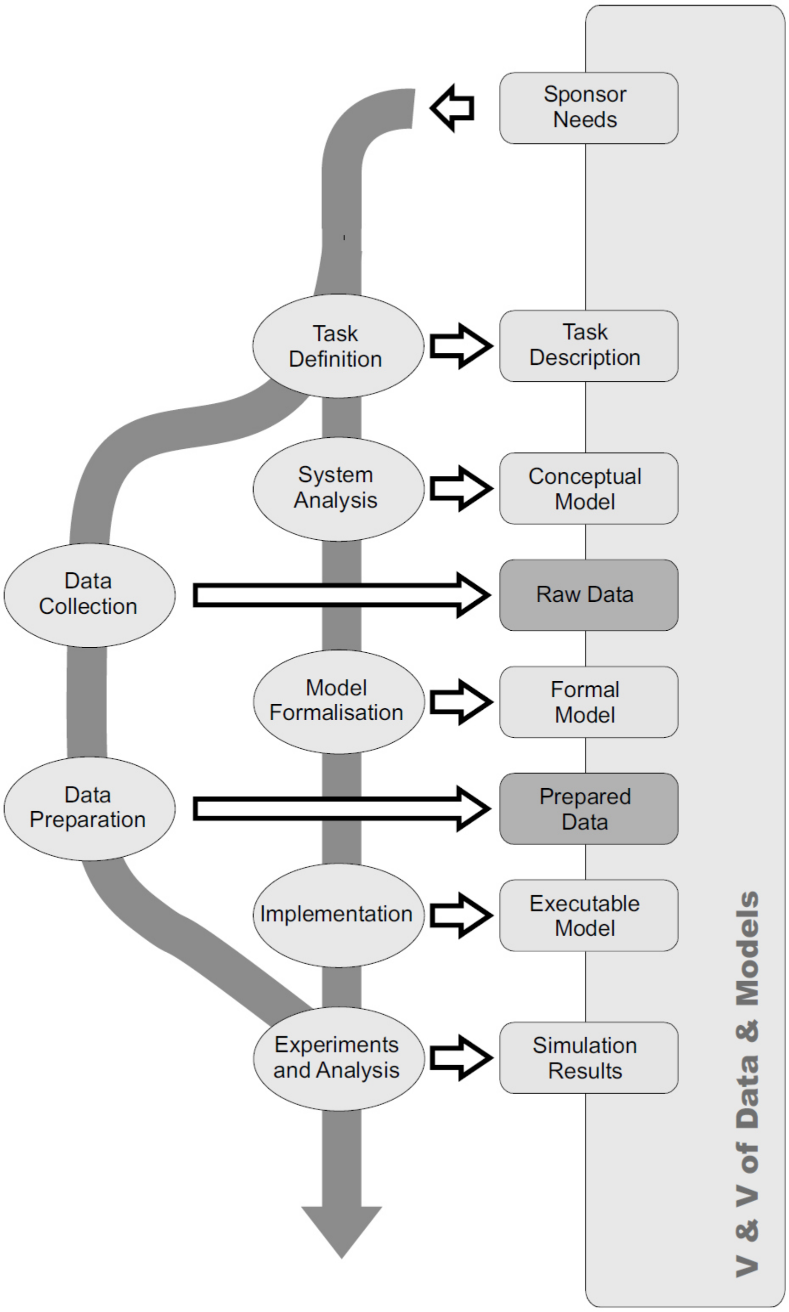

Figure 9: procedure model for simulation including V&V[69]

Starting from a defined model purpose and objective, the V&V procedure model considers tasks which are handled within a simulation project after a simulation study was assigned. In contrary to other procedure models (stated in Rabe et. al. 2008b), the V&V model focuses on five phases and their results and splits up the handling of data

[69] Rabe, M. et al. (2008a), p. 1720.

and the model creation process itself. The phases data collection and data preparation as their results raw data and prepared data are excluded from the sequence of modelling steps because they can be handled separately aside the modelling process. Due to the complex character of most simulation models, V&V must be a constant doing during the model creation process which is underlined by the big vertical task on the right of Figure 9, instead of performing V&V only after finishing a model. Verification and validation are performed as tests on task results instead of tasks themselves - which is also visualised in the procedure model.[70]

Validation refers to the evaluation if a model is correct in respect of behaviour and representational accuracy. This is especially relevant when breaking down the reality into a conceptual model in the first place, thus also called **conceptual model validation**. Secondly, **operational validation** is determining if the output of an executable computerised model has sufficient accuracy.[71]

Besides validation, the model designer must also ensure a verified model. **Verification** describes the correctness of applied rules, functionalities and components, Balci (2003) summed it up as 'Are we creating the X right?'. Verification is not about a correct behaviour of a model, but rather about the correct transformation of the conceptual model into a computerised environment.[72]

A valid model on the other hand matches the real system regarding all relevant characteristics including the temporal behaviour of reality, as formulated by Balci (2003) 'Are we creating the right X?'. Validation approves the suitability of a model for a given task or purpose. Hedtstück (2013) sums up the difference between those two terms by the sentence: Verification expresses that the model has been designed correctly, whereas validation means that the correct model has been setup. It should be noted that neither correctness nor suitability can ever be completely granted, that's why a systematic V&V approach should always be applied in order to ensure credibility of stakeholders and users into a model.[73]

Having a task description defined (model's purpose), the next step is to analyse the inspected system and draft a **conceptual model** from those insights. Such a model is symbolic and not directly executable, but it describes system boundaries and relevant properties that came into the modeller's mind while observing the real system. Typical involved techniques are textual system descriptions, diagrammatic semi-formal descriptions such as Petri nets, value stream maps (VSM) or UML or measured/estimated model parameters and input distributions.[74]

Building on the outcome of the system analysis and the resulting conceptual model, in the following phases model formalisation and implementation further steps are taken in order to accomplish an executable model. A **formal model** includes preparations for the upcoming model implementation, providing a formal design which is later transformed into a coded digital programme with the help of a software tool. Documents which were

---

[70] cf. Rabe, M. et al. (2008b), p. 4 pp.
[71] cf. Page, B.; Kreutzer, W. (2005), p. 195 p.; Balci, O. (2003), p. 150; Sargent, R. (2010), 168.
[72] cf. Rabe, M. et al. (2008b), p. 14 p.; Hedtstück, U. (2013), p. 8; Sargent, R. (2010), 168; Balci, O. (2003), p. 150.;
[73] cf. Hedtstück, U. (2013), p. 8; Rabe, M. et al. (2008b), p. 15 p.; Balci, O. (2003), p. 150.
[74] cf. Page, B.; Kreutzer, W. (2005), p. 212; Tempelmeier, H. (2018), p. 23.

used to formulate the conceptual model get extended and additionally decisions should be made regarding assisting software regarding data bases, spreadsheet calculation and interface management. At this stage regularly mathematical equations are formulated, describing the real system in an abstracted and simplified way. Further in use are software engineering techniques which support the creation of transparent programme code such as ER-diagrams, decision tables and identifying re-usable model components. The nine V&V criteria completeness, consistency, accuracy, currency, applicability, plausibility, clarity, feasibility and accessibility should be considered constantly.[75]

The phase **data collection** relates to tasks that include the actual acquisition of needed inputs including the definition of data entity types and their attributes as well as data types. The term data entity type refers to a group of similar model elements (entities), which can be described by common attributes. In production and logistics processes usually already a big variety of data exists and gets created permanently. Usual data entities are working plans, inventory levels or production orders for example. The more challenging task is to get a valid data set without incorrect values from wrong sensors, human error or calculation mistakes. Inconsistencies and mistakes found during the collection and preparation phase should be documented to avoid them next time. In order to get **prepared data** which can further be utilised in a model, data inputs probably will not be available in the needed input format or granularity. Thus, often additional data needs to be generated, converted, filtered or processed to have information available as needed. [76]

Barton and Szczerbicka already stated in 2000 the following typical errors that lead to invalid models. Firstly, wrongly or too drastically simplified components or functionalities need to be avoided, as also imprecise estimates of parameter values due to missing data. Secondly, invalid or insufficiently accurate input data of an approximated real system's behaviour lead to bad models. Moreover, errors need to be prevented which are introduced during the conversion of a conceptual model (draft in the modeller's brain) into an actual software programme.[77]

The realisation and implementation of an **executable model** results by enhancing the existing conceptual and formal model into a digital representation with the help of a simulation tool. Such software tools allow the reproduction of a system in the manner of a computerised model. The variety of such tools can be split into the following classes:[78]

- Level 0: pure programming languages (implementation language) without any pre-defined components/building blocks
- Level 1: programming languages including simulation-relevant basic components
- Level 2: general simulation tools
- Level 3: specialised tools for certain application areas including specially designed model elements for e.g. production and logistics

[75] cf. Tempelmeier, H. (2018), p. 23 p.; Page, B.; Kreutzer, W. (2005), p. 215; Rabe, M. et al. (2008b), p. 74 p.
[76] cf. Rabe, M. et al. (2008b), p. 86 pp.
[77] cf. Page, B.; Kreutzer, W. (2005), p. 196.
[78] cf. Tempelmeier, H. (2018), p. 16 pp.; Rabe, M. et al. (2008b), p. 78.;

- Level 4: software specialised for parts of certain sub-areas of application fields including more pre-designed model elements for that specific sub-area

General simulation tools of Level 2 (including Level 0 and 1) were not developed on a focused area of use like production systems for example and can be applied in various fields. However, their level of complexity usually requires long training periods and often demands the involvement of specialists. More specialised tools are targeted for issues of their area of application and have corresponding characteristics in their functions and model elements such as pre-defined building blocks where only a view variables have to be set. Examples in the area of production and logistics are material flow optimisations, personnel planning, (driverless) transport systems or line feed processes. Such software solutions are often known by the name ‘simulators’.[79]

Having designed a verified and valid model (more detailed mentioned in chapter 2.4.4 V&V Techniques), the final and desired task can be performed: running experiments and analysis which should lead to meaningful **simulation results**. Nevertheless, meaningful interpretation of those results and the derivation of measures for the investigated system are only possible if the experiment parameter variations are targeted and systematic. The result interpretation always takes place in cooperation between the planner or customer together with the simulation expert. The processing of results for interpretation should be performed during or right after every simulation run. Categorically, it is distinguished between tabular and graphical presentation of cumulative results including the visualisation of progress data. The choice of graphical presentation has to be determined for the wished simulation experiment statement individually, nowadays a great selection of line, column, circle, pie or Sankey-diagrams is available in various libraries.[80]

A very descriptive way to visually explain the range of a dataset is a boxplot, sometimes also called whisker plot, shown in Figure 10. It summarises the range of data by 5 statistical measures ‘minimum’, first quartile Q1, median, third quartile Q3 and ‘maximum’. The median is a robust measure to outliers by representing the middle value of a dataset, meaning that 50% of all values are bigger and 50% are smaller than itself. The first quartile Q1 or 25th percentile represents the middle number between the smallest and the median number. The third quartile or 75th percentile is the middle value between the median and the highest value of the dataset. Subtracting Q1 from Q3 results as so-called interquartile range (IQR). Q3 + IQR*1.5 reflects the ‘maximum’ of the dataset, meaning that all values bigger can be interpreted as outliers. Q1 – IQR*1.5 shows the regular ‘minimum’ of a dataset, smaller values represent outliers again. Figure 10 visually presents the mentioned measures and compares them to a nearly normal distribution’s probability density function.[81]

[79] cf. Tempelmeier, H. (2018), p. 16 pp.
[80] cf. Rabe, M. et al. (2008b), p. 81; Tempelmeier, H. (2018), p. 26 p.;
[81] cf. Molin, S. (2019), p. 19 pp.; Galarnyk, M. (2022).

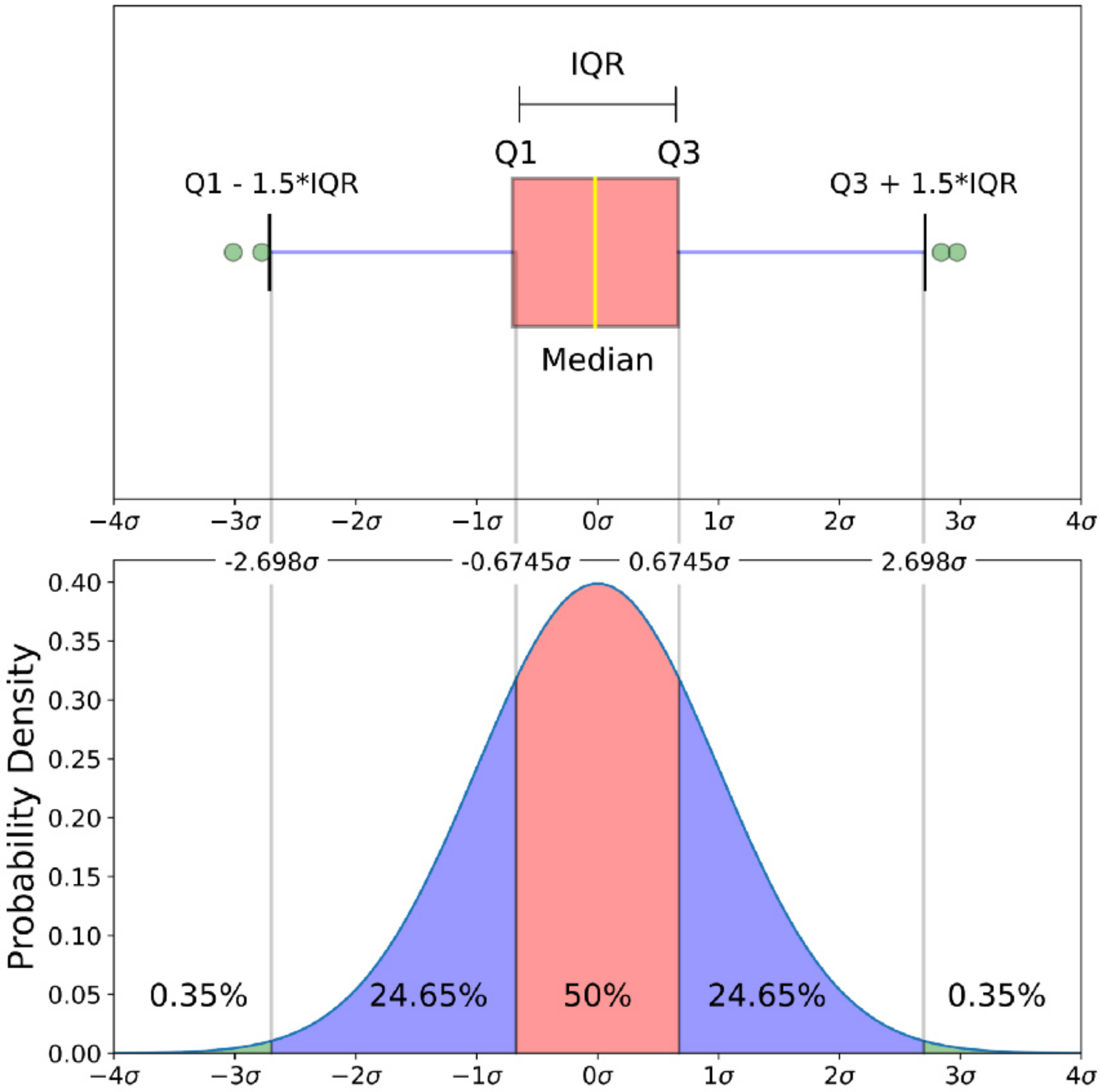

Figure 10: boxplot of a nearly normal distribution[82]

Various authors have developed further possibilities of complex data display options, for example Linda Gustafsson-Ende conducted a visualisation study at BMW AG with the title 'A visualization concept for production data and simulation results' and created a generic framework for visualising production data. Other remarkable visualisation authors active on various streaming platforms are David Kriesel or Valentina D'Efilippo.

2.4.4 V&V Techniques

Achieving credibility and having trust into the model is probably number one success criterion of every simulation project. That shows the importance of insuring model validity, confirming the necessity why the validation process should already start when setting up a conceptual model. Ensuring that the model mirrors a plausible representation of the real system which is suitable to answer all questions raised in the problem statement. Rabe underlines this procedure, stating that regular tests should

[82] Galarnyk, M. (2022).

determine if the model represents the examined system reasonably accurate, includes the defined functionalities and meets the technical requirements (such as computing time). Those test results should be documented in form of reports after they have been performed. In case of negative test results, affected results of prior tests have to be revised.[83]

Important exemplary validity steps throughout the whole modelling cycle are:[84]

- Determine the collected data quality of the real system before using it and avoid the known problem statement 'garbage in – garbage out.' This can be done by investigating measured data or the measurement process itself or use more empirical research methods like statistics or compare the data to existing theories.
- Observations or any informal conceptual model derived from theories must get validated against data and other relevant theories.
- The transformation process during the formal design of a conceptual model and its final implementation must be verified. Pilot runs should be conducted for operational model validation before actual experiments get analysed. Such runs ensure the model behaviour is checked for plausibility and compared to available reference data of the real system. If model results don't fit to real systems behaviour after the first iteration (which is a common issue), model calibration is an important aspect of operational validation. This involves the adjustment of model parameters in order to improve model predictions.

Further exemplary model validation techniques include the following, a more detailed list can be found in Rabe et al. (2008b):[85]

- Statistical Techniques (also called quantitative techniques): used to evaluate the level of certainty with which simulated output variables follow the behaviour of a real system. Additionally, these techniques can determine the validity of used distributions for input parameters such as downtimes and can be split into:
    - Adjustment tests which specify, if used distributions reflect the reality sufficiently. Most common is the Chi-square test which can be described as a comparison of a histogram and a distribution function. Other examples are the Kolmogorov-Smirnov or Anderson-Darlin test.
    - Hypothesis or parameter tests (rejection or acceptance tests) examine if simulation results of an experiment are sufficiently accurate regarding the real system or not.
- Extreme-Condition Test: the model behaviour is tested regarding an expected outcome if model parameters are set to an extreme level. Applying this test, errors will come up regarding missing or wrong conversion factors, invalid process times can be discovered, or the probability of wrong parts can be set to zero, in order to test if no rework takes place.
- Sensitivity Analysis: input parameters are changed and the impact on output parameters is analysed. The effect direction of change must correspond to known behaviour of the real system. 'Sensitive parameters' have to be observed especially critically, meaning that small changes can have big impacts on result

[83] cf. Rabe, M. et al. (2008b), p. 16; Page, B.; Kreutzer, W. (2005), p. 199 p.
[84] cf. Page, B.; Kreutzer, W. (2005), p. 201 p.
[85] cf. Rabe, M. et al. (2008b), p. 95 pp.; Gutenschwager, K. et al. (2017), p. 209 pp.;

variables. An example would be the sickness ratio, shift time or wrong choice of downtime distributions for example.

- Structured Walkthrough: originally a software development technique, project participants meet up and go through every section of the model or the respective model documentation together.
- Internal Validity Test: This test is based on a stochastic model, where several simulation runs are compared to each other without changed parameters but different random seed numbers. A significant deviation of results can lead to two different conclusions: On the one hand, the model can result as not valid if the real system isn't facing such big variations. On the other hand, the model could correctly represent the actual range of variation, it then must be questioned whether these fluctuations are acceptable for the behaviour of the real system.
- Trace Analysis: the behaviour of single model objects is examined by recording remarkable events in a trace-file, checking for their logical attitude and plausibility after the simulation run.
- Animation: a very common and liked method for visual-oriented user, where animation can be used to analyse certain situations and show that a model is not valid, but a 'correct' animation is no guarantee of a valid or debugged model. Conclusions about temporal processes can be represented in a good way and compared to differences in the real system, but conclusions beyond this behaviour are hardly possible. Errors in the model that rarely occur are very unlikely to be detected.
- Historical Data Validation: can be used if a real system is modelled and if historical input and output data is available. The data is split up into parts, one part is used to train the model and the separate part is used to validate the already trained model afterwards.

If further approaches are desired, more than 77 V&V techniques can be found in the simulation handbook of Banks (1998), written by Osman Balci.[86]

Statistical techniques have a strong mathematical focus. For a better understanding, their approach is discussed in more detail: A statistical hypothesis is a statement or assumption regarding the distribution of a random variable or the result of a simulation regarding the reality. The analysed assumption is called null hypothesis H_O, the comparative statement as alternative hypothesis H_A. The result of the statistical test is either by accepting or rejecting the null hypothesis. A test rejects the null hypothesis if the random variable or simulation result is located in the so-called rejection area. Rejecting the null hypothesis even though it is correct, is called a first kind error and the probability of making such an error is called α. Accepting the null hypothesis even though it is wrong, is called a second kind error and the probability of making such an error is called β. Common values for those metrics are $\alpha \leq 0.10$ and $\beta \leq 0.10$. A probability of α is also called significance level. The so-called power of test or p-value is the probability of rejecting H_O, when the hypothesis is wrong, which is $1-\beta$.[87]

In order to prove that simulation tests are valid, chosen parameters need to be compared to the reality and accept the test if the p-value is ≥ 0.1. The used test should compare two different and independently mean parameters (reality and simulation result) between

[86] cf. Banks, J. (1998), p. 335 pp.
[87] cf. Chair of Mathematics and Statistics (2015), p. 37 p.

each other. A possibility to test this requirement is the two-dimensional t-test, assuming that the variance of both samples is equal. This test is used to decide if the null hypothesis can be rejected. A relevant null hypothesis for simulation results is that the compared metrics (reality vs. simulation) have the same mean-values: $H_O: \mu_1 = \mu_2$, the alternative hypothesis would then be, that they are unequal: $H_A: \mu_1 \neq \mu_2$. The null hypothesis has to be rejected, if the test result is in the rejection area.

Additional statistical validation metrics that are used in terms of evaluating modelled results are:[88]

- Mean Absolute Error (MAE): A measure containing information about the difference between the real and the predicted value over all recorded samples, where y is the actual value and $\hat{y}$ the predicted value. It measures the average of residuals (delta between y_i and $\hat{y_i}$) in a dataset and indicates the average error that can be expected about the predicted values. 0 would mean that there is no error in the predicted values, the lower the MAE value is, the better the model result is.
  - $MAE = \frac{1}{n}\sum_i |y_i - \hat{y_i}|$
- Root Mean Squared Error (RMSE): As square root of the Mean Squared Error (MSE), it is a metric which measures the standard deviation of residuals, it further penalizes poor predictions.
  - $RMSE = \sqrt{MSE} = \sqrt{\frac{\sum_i (y_i - \hat{y_i})^2}{n}}$
- R-squared measures the variation that can be explained in the model, for example the predicted WIP of a production environment. It is also called the coefficient of determination and is a scale-free metric between 0 and 1. The higher resulting value (percentage), the better model predictions will be. $\bar{y}$ indicates the average of the actual values y.
  - $R^2 = 1 - \frac{\sum_i (y_i - \hat{y_i})^2}{\sum_i (y_i - \bar{y})^2}$

By the definition of different simulation methods, detailed explanation of DES and a discussion about a procedure model of simulation including various verification and validation metrics, the chapter covering simulation basics is finished.

2.5 Simulation Implications for Production Systems

Having discussed the scope of simulation in the previous pages, this last theoretical chapter gives insights about current research projects striving to apply and generate added value by combining all topics of the previous chapters in section 2.

The term digital twin (DT) has gained more and more attraction over the past years, mainly because it is seen as core enabler of smart manufacturing. Further empowering focus technologies around smart manufacturing include the Internet of Things (IoT) - boosting sensors' communication capabilities, cloud computing (CC) - shifting server capacities from local on-premises models into the cloud, big data analytics - extracting knowledge out of the tons of data created and artificial intelligence (AI) - designing smart

[88] cf. Molin, S. (2019), p. 551 pp.; Acharya, S. (14.05.2021).

algorithms that should help in various use-cases. Combining these technologies develop the current knowledge-based manufacturing practices to more data-driven and knowledge-enabled smart manufacturing procedures. In such a decentralised system, machines and other connected process resources will be able to make decisions based on the current state of the system. Such a decentralised system is also called cyber-physical system (CPS).[89]

The difference between DT and CPS can be described as a CPS focuses on computing, communication and control, providing real-time sensing, information feedback and dynamic control. Sensed data are used to monitor and control physical entities in a safe, reliable, collaborative and efficient way. A DT concentrates more on mapping the real world in a validated model of physical objects, simulating the behaviour including additional input parameters as also historical data and provides feedback as further predictions to the real world. The complete lifecycle of system's participants including a complete digital footprint is more centric compared to CPS. In terms of an evolving smart factory applying smart manufacturing technologies, the two terms will emerge simultaneously and will make it difficult to draw a clear line between the two.[90]

Industry 4.0 is a related term, declared by the German government in 2011, as announcement of the fourth industrial revolution. The realisation of this concept also includes the enabler technologies of smart manufacturing in order to realise smart factories that enhance flexibility and optimisation of resources to provide better customer service. The essential approaches cover digitalisation of processes to allow immediate communication and information exchange in real-time, the use of simulation and data processing in order to generate knowledge out of collected data and improve energy and resource efficiency. By applying those proposals, the industry in particular expects cost benefits by means of less inventory costs because of less WIP and reduction of wrong order amounts, more productivity by reducing breakdowns and idle time and less logistics cost by increasing the level of automation regarding transport and picking. Additionally, complexity and quality costs should go down because of transparent traceability data and faster quality determination processes and maintenance costs should be reduced due to predictive maintenance enabled by smart sensors.[91]

The term smart factory in general implicates a fully interconnected and intelligent manufacturing site of the future. It is characterised by a new level of integration regarding human-technology interaction along all involved actors and resources within a production system in real-time. The central target state behind this term is a network of autonomous and spatial self-steering, self-controlling, self-configuring, knowledge-based, sensor-supported and autonomous production resources such as operators, machines, robots, warehouse systems and conveyors including planning and control systems. Combining the already established approach of lean manufacturing together with a digital factory, like defined in the introduction of chapter 2.4, results in the concept of a smart factory.[92]

[89] cf. Tao, F. et al. (2019), p. 653; Zhang, H. et al. (2022), p. 417.
[90] cf. Lee, J. et al. (2020), p. 34; Tao, F. et al. (2019), p. 653 pp.
[91] cf. Pawlewski, P. et al. (2021), p. 1 pp.; Bauernhansl, T. et al. (2014), p. 31.
[92] cf. Bracht, U. et al. (2018), p. 429.

A central question for companies that are aware of the necessity to adapt to the mentioned concepts and topics is 'How to proceed?'. Bauernhansl recommends to start and realise use cases, that are most beneficial for each individual application, focussing on activities that at least support or empower value-adding processes.[93]

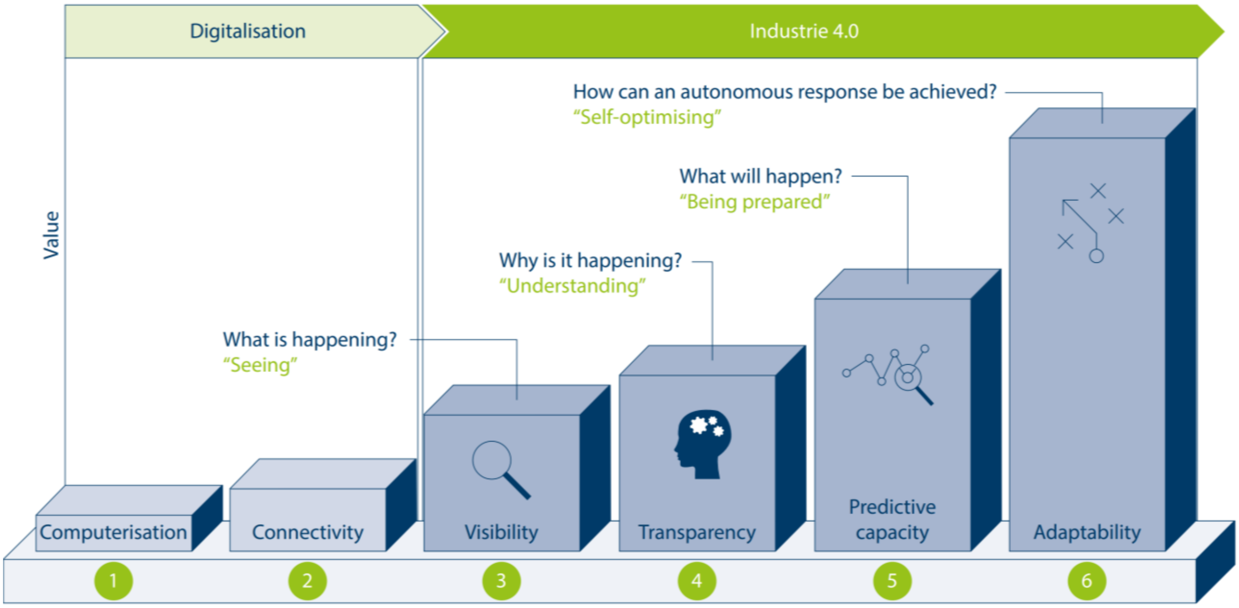

Figure 11: development stages in Industry 4.0[94]

Important to note are the different stages in context of achieving a smart factory, shown in Figure 11. Computerisation (1) and connectivity (2) are defined as preliminary stages and foundations for Industry 4.0 in order to know what is happening, workplaces, orders and other process participants have to provide digital information. Knowing how to access these data is step 1 and described as visibility (3) to be aware what is happening. A resulting structured database system that collects, maintains and provides needed data to describe the status-quo implicates a so-called digital thread. Setting this data into context and know why processes or events occur is summarised as transparency (4). Further developing the gained knowledge into a computerised model allows the possibility to predict (5) incidents based on historical data combined with real-time information. If there exists an automated one-way data flow from the real world into the virtual entity, researchers describe such a model as digital shadow. A fully developed model has finally the possibility to optimise parameters itself which is presented as adaptability (6). Realising a bi-directional data flow which automatically exchanges information between the real and the virtual world can be defined as DT.[95]

As mentioned in the introduction, there exist various definitions about digital twins focussing on different aspects. The currently most cited definition of Tao includes five

[93] cf. Bauernhansl, T. et al. (2014), p. 32 p.
[94] Schuh, G. et al. (2020), p. 18.
[95] cf. Joppen, R. et al. (2019), p. 760; Kritzinger, W. et al. (2018), p. 1016 p.; Kaiblinger, A.; Woschank, M. (2022), p. 12 pp.

required dimensions regarding physical objects, a virtual model, data, a service system and connections between the previous four.[96]

Such a bi-directional real-time interaction example represents a real-time scheduling (RTS) model. Thereby, current production data are used to create and update production schedules due to unexpected order or material arrivals or a sudden change of resource availability like a machine breakdown. Such models are especially designed for dynamic and stochastic flexible job-shops.[97]

Summarising, simulation in general is a core discipline needed to deploy future relevant concepts in order to combine the variety of input factors, analyse different scenarios and act as decision support. Even methodologies like Lean Management have already discovered the advantage of applying simulation methods by evaluating the impact of lean initiatives before roll-out in production systems for example.[98]

Having understood the principles of a value adding processes, applying the correct combination of key performance indicators and constantly identifying bottlenecks including their optimisation enables successful value creation in production systems. Using the power of simulation correctly on top of that enhances the potential to determine competing relations or extreme scenarios before they actually happen and to prevent them by rescheduling.

This chapter closes the section theoretical fundamentals of this thesis. In the following section 3, gained knowledge of the previous pages will be used to identify bottlenecks with the help of simulation in a case study applied to a job-shop of an industry partner.

[96] cf. Kaiblinger, A.; Woschank, M. (2022), p. 8 p.
[97] cf. Ghaleb, M. et al. (2020), p. 2.
[98] cf. Shou, W. et al. (2021), p. 2120.

3 Empirical Exploration

In chapter 2.3 different bottleneck detection methods are listed that are nowadays used. Out of all mentioned, simulation is chosen for this thesis, because a dynamic system behaviour should be modelled, no data regarding material transports is available and a job-shop environment is present. Additionally shifting bottlenecks should be detected in combination with a varying product mix and volume. Gained knowledge from the cited approaches active period and real-time bottleneck detection will be covered regarding focussing on workplace occupancy rates and queue waiting times. A developed DES model gets validated regarding that target by means of a case study at an industry partner’s job-shop including 43 workplaces that handle 9 different operation technologies. The collaborative and dynamic shopfloor follows the make-to-order (MTO) principle and FiFo queues are deployed. The realised model will then be utilised to evaluate how simulation can be used to identify bottlenecks in a dynamic system of planned future scenarios. Gained knowledge of the theoretical fundamentals is applied and tested on an existing workshop production environment.

First, the cooperation partner and considered production system are briefly explained, followed by the performed steps aligned to the mentioned V&V procedure model of simulation, described in chapter 2.4.3.

3.1 Industry Partner and considered Production Environment

The partner company acts as international technology and mechanical engineering company and is globally leading in its core business as manufacturer and provider of innovative crane and lifting solutions. With more than 12.000 employees and over 30 production sites, the company has a big focus on leveraging digitalisation along its internal processes. Production planning follows the MTO principle, executing several MRP runs for the global production network. Based on those calculations, it is the daily work of the logistics departments, to release production orders matched to available resources to the shopfloor. In order to assist this task, a new scheduling software roll-out is planned for 2023, where the characteristics and sorts of bottlenecks needs to be defined. Knowledge development regarding that requirement is one expected target from the industry partner towards this thesis.

Generally important to note is, that observations presented in this thesis do not allow any direct conclusions about the company's production situation, as only a limited area including limitations (listed in chapter 3.4), and not the overall value chain network, was analysed.

The examined production environment of this thesis is chosen in cooperation with the industry partner and finally a manufacturing site in Europe is selected which is specialised in the production of processed and welded metal sub-assembly parts. The determined system boundaries reach from the overall value stream start characterised by laser and autogenous cutting of metal sheets or sawing of metal tubes until the last processing step before tacking for welding. Operation steps taken into account include chamfering, turning, deburring, drilling, milling, straightening and bending. Such a technology combination can be summarised as classical job-shop fabrications area with all known scheduling challenges determined behind the term job-shop scheduling problem (JSSP). The level of automation in the respective workshop area is due to the big product variance and the amount of different handled technologies quite low, varying between the different mentioned technologies. In total 43 workplaces are considered, a more detailed description is shown in Table 2, additionally a visualised layout of the covered 5.500m² can be found in Figure 12, cited in chapter 3.2. Generally spoken, the regular number of shifts varies between 2 and 4 shifts. Nevertheless, workplaces which are not working in 4 shifts do overtime shifts on the weekend, if the current workload or backlog is high.

Due to a new scheduling software roll-out planned for 2023, this thesis evaluates the characteristics and influence factors of existing and future bottlenecks.

Having that defined, the first phase of the procedure model and its result as **task description** is fulfilled.

3.2 Conceptual Model

In the next phase, a system analysis is performed by accompanying a three-day value stream design (VSD) workshop in summer 2022. This workshop starts with summing up the value stream map (VSM) workshop that has already taken place. During the next three days future initiatives, planned production volumes, workplace and layout changes are discussed. These inputs are consolidated into a standardised spreadsheet and capacity shortages determined statically on a yearly basis.

Valuable knowledge is gained regarding existing processes, planning procedures and current bottleneck handling methods. The used value stream map represents a very good source of information, to follow discussions and get familiar with the production environment's value stream. Due to data sharing policies of the industry partner, the process flow diagram cannot be shared in this thesis. Besides the workshop, additional time is used to observe the material and information flow live and talk to local employees about established procedures and common practices. An important topic to clarify is for example, how long it takes that finished operations are booked into the ERP system to know how accurate those timestamps are.

During the stay the actual system boundaries are determined and all finally considered workplaces (WPLs) are listed in Table 2, an overview about the inspected production environment consisting of about 5.500m² can be found in Figure 12.

WPL type (abbreviation)	nr. of WPLs	operators/WPL
Laser cutting (MYXL)	7	2
Autogenous cutting (MYX)	1	1
Sawing (MIB)	2	1
Chamfering (MSB)	7	1
Turning (MDC)	5	1
Deburring (PSFB)	4	1
Drilling (MBR)	5	1
Milling (MFC)	4	1
Straightening (MRB)	2	1
Bending (MK)	6	2
sum	43	56

Table 2: workplace overview

In general, the value stream of the analysed production environment can be described as streaming from the right side on Figure 12 to the left. Production orders for metal sheets either start, depending on their thickness, at one of the laser machines (MYXL) or on the autogenous workplace (MYX) coloured in light green. Those parts afterwards generally go to deburring and then to chamfering workplaces (yellow in the middle), followed by various additional working steps including drilling, milling, straightening and bending. The other big value stream affects pipes, which start with sawing in the top right corner and afterwards get processed at turning machines, coloured in light turquoise next to chamfering in the middle.

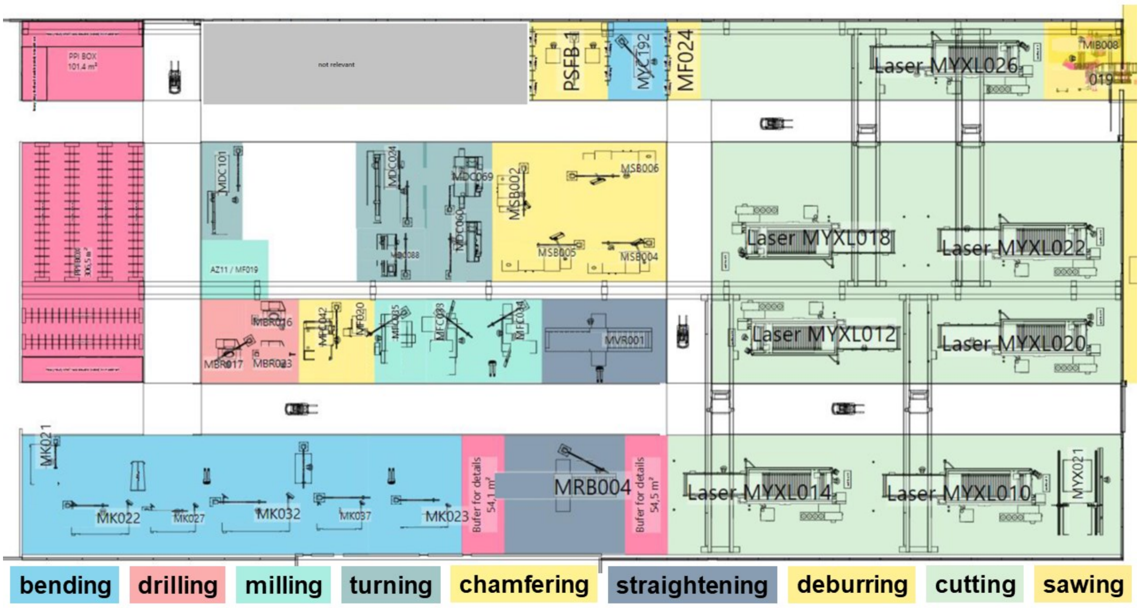

Figure 12: analysed production environment layout[99]

[99] industryPartnerExperts (2022).

Spending time at the plant locally, additionally allows to gather needed data inputs from process experts, such as workplace lists, shift models, logged team attendance times and material specific planning know-how to understand data patterns better. Furthermore, information exchange is possible about performance levels on certain workplaces or know-how about the capabilities and limitations of different workplaces.

The second phase system analysis is finished by acquiring all listed information mentioned above including workplace descriptions, layout and VSM about the considered production environment as conceptual model, which gets validated by the local process expert team.

3.3 Data Collection and Preparation

This chapter is all about performed data collection and done preparation to finally use the needed data in the DES model. Like mentioned in the procedure model in chapter 2.4.3, these two tasks can be performed aside the modelling process. In order to get an overview of workplaces, layout and handled order types, this phase was already started before accompanying the local workshop. Especially workplaces and the number of shifts per workplace are discussed in detail with a process expert during the local stay.

Based on those interviews, the analysed time period is set to January 2022 until end of June 2022. The first two months act as so-called warm-up or transient phase for the simulation model where queues and WIP is building up until the system reaches a realistic level. This is a common practice, described in more detail in chapter 3.5.2 and visualised in Figure 20, if no real-time data regarding current inventory amount or queue-length is available. In the total six months, over 130.000 production orders with more than 7.600 different material numbers are simulated. These orders include between one and nine different handled operations.

The following subchapters are split regarding all used data sources that are gathered and afterwards prepared in the described way.

3.3.1 Orders and Scenarios

Operations resulting from production orders represent the core input for DES in a job-shop, if resulting bottlenecks at defined workplaces are tried to be found. In order to develop a valid simulation model, all booked operations between January and June 2022 are exported out of the ERP system. This data set is used to create and validate a DES model which is afterwards used to evaluate future scenario operations of 2023. The term ‘orders’ is used for the respective time period in 2022, a ‘scenario’ is the same data entity, but for a 6-month period in 2023. Both data entities include the following relevant **attributes**: order number, order quantity, order unit, material, planned end and start date, setup time and unit, process time and unit, operation number, operation description, respective workplace and booked timestamp.

Like mentioned in chapter 2.4.3, it is necessary to ensure that used data is available in the needed data type. Based on the input txt-file a simulation user can define a couple of configurations that will be considered when importing raw data and saving it as processed and ready to use Parquet-file. That includes:

- Input data name: the file name including the purpose of the file has to be defined. Different implemented purpose types are orders, workplaces, WIP, downtimes and scenarios.
- Input data type: possible input types are .csv with delimiter ';' or .xlsx.
- Convert columns to specified data type: in order to have processed data available in the needed data type, it is possible to define for every input file the wished data type per column. Pandas, described in chapter 3.4, often recognises the correct type, but it turns out that if outliers or unwanted line breaks are included in input files, wrong types are assigned. The implemented converting possibilities are:
  - Change string dates from format YYYYmmdd to date format.
  - Convert dd.mm.YYYY to date format.
  - Accept datetime format as datetime.
  - Combine two columns (e.g. date and time) in order to receive datetime.
  - Remove blank spaces of any String column.
  - Combine two String columns into one.
  - Define a column as numeric in order to receive a decimal (float64) or a whole (int64) numbers.
- Define columns that must be included in every row, otherwise the row won't be accepted.
- Apply calculations or filters to specified columns to adapt wrong or not available exported values. This involves for example the adaption of units or only import relevant values needed in the simulation. Those possibilities include:
  - Divide all column values by a certain number.
  - Add a new column by dividing, multiplying, subtracting or adding two other columns with each other.
  - Add a new column by dividing, multiplying, subtracting or adding a certain number to an existing column.
  - Delete a specific column.
  - Filter for certain dates and only continue with rows within that period.
  - Filter for values of another file, for example only continue with operations which are relevant for workplaces (WPL) in the defined WPL-input file.
  - Filter for rows containing a certain value.

All those configuration possibilities are mainly implemented to cover not ideal raw data quality of orders or scenarios. Nevertheless, all developed adaption formulas can be applied to all other input file types. Finally, the converted order data is validated by expert reviews.

3.3.2 Workplaces

Another important input for a DES of a production environment are all considered workplaces. First, the involved workplaces are exported out of the ERP system including the amount of defined shifts. Afterwards, this list is extended regarding additional needed information like number of operators, performance rate and shared queues. At the end, the following attributes are used by the simulation model:

- WPL ID: unique ID of the ERP system for each workplace.

- WPL name: an alpha numeric workplace abbreviation like MYXL including a number for lasers for example.
- Plant number: ID of the workplace's plant location.
- WPL description: description including the technology and machine manufacturer.
- Team: every workplace is assigned to a team number which is an abbreviation for the WPL group and mainly used in the ERP system.
- WPL group: every workplace is additionally assigned to a technology group like cutting for lasers or bending for press brakes.
- Number of shifts: the number of shifts varies between 2 and 4 shifts. Shift time durations including breaks are defined in the txt-input file.
- Number of workplaces: needed for some of the workplaces if there exist more than one workplace, like manual deburring workplaces for example.
- Number of operators per workplace: some workplaces require 2 operators, those can be identified in Table 2.
- Performance rate: some workplaces have a higher standard performance, mainly because working steps are included in the working plan, but they are not necessary for all respective parts, an example for that case would be straightening. Standard times will be decreased by that factor in the model.
- Queue 1: the model provides the opportunity to define explicit queues for each workplace. If no unique queue is defined, the workplace is added to its technology-queue. This queue logic is explained in more detail in chapter 3.5.1.
- Queue 2: this attribute gives the possibility to define alternative workplaces by assisting other workplaces.
- Comment: this attribute is especially implemented for workplaces that are added to the production environment during the simulation period and further explained in chapter 3.5.1.

All listed attributes are maintained for all 43 workplaces in an Excel input list, which is imported and saved as Parquet-file and as such used from the simulation model.

3.3.3 WIP

As WIP has a significant impact for the cooperation partner and as it has big influence on the lead time in general, like mentioned in chapter 2.2, WIP will be an evaluated KPI in this thesis. The plan is to compare the existing WIP of the reality in 2022 versus the simulated amount of the same period. For this purpose, a report is exported of the ERP system that lists all different storage location bookings. Matching this data with production order bookings, it is possible to reproduce the time material entered and left the production environment. Based on the given data situation, only finished operation timestamps are available (retrograde booking), thus no WIP and lead time can be determined for production orders with single operations. The amount of WIP only includes the raw material value, meaning that no added value is taken into account if processed at several workplaces. Regarding data preparation it is necessary to correctly exclude cancelled bookings from the dataset otherwise wrong WIP levels are considered. In order to avoid further data related concerns, production orders that have no final booking into a finished goods-storage location get excluded as well.

3.3.4 Breakdowns and Maintenance

A production system is not always in a 100% available processing state. Machines break down, tools and fixtures crack or service is necessary to be applied. To design a model that represents the reality, breakdowns of the past 18 months reported from the maintenance team in the ERP system are exported for every available workplace. Due to the importance to the local management team, also short downtimes of only half an hour are reported there, thus the validity of those datasets is given.

Two relevant durations are examined based on the reported timestamps and length of repair times due to **breakdowns**: Firstly, the time period which passes after a machine breaks down, also known as time-to-repair (TTR) and the probably known KPI mean-time-to-repair (MTTR). The second relevant time period is time-between-failure (TBF) and its KPI mean-time-between-failure (MTBF) which considers the mean operating time between two failures. According to Gutenschwager (2017), the so-called stationary availability KPI (*German: 'stationäre Verfügbarkeit'*) is defined in the standard DIN 40041 (1990, p. 8) by the following formula:[100]

$$stationary\ availability = \frac{MTBF}{MTBF + MTTR}$$

It should be mentioned that some authors consider the mean-time-to-failure (MTTF) as the previous stated definition of MTBF, which is contrary to the IEC Norm 271 that defines MTTF as the mean operating time for non-reparable products.[101]

Since MTTR and MTBF are mean values and assume a standard deviation for both value sets, it is discovered very fast that both duration sets do not follow a standard deviation. After some research it can be stated that literature assumes an exponential distribution for TBF, most also assign TTR as exponential, others cite Erlang, log-normal, alpha or Weibull distribution as appropriate.[102]

WPL type (abbreviation)	consideration	best TTR distribution
Laser cutting (MYXL)	stochastic	non-central t-distribution (nct)
Autogenous cutting (MYX)	deterministic	-
Sawing (MIB)	stochastic	non-central f-distribution (ncf)
Chamfering (MSB)	stochastic	ncf / alpha
Turning (MDC)	stochastic	ncf / exponentiated Weibull (exponweib)
Deburring (PSFB)	none	-
Drilling (MBR)	stochastic	exponential (expon)
Milling (MFC)	stochastic	expon / exponweib
Straightening (MRB)	deterministic	-
Bending (MK)	stochastic	gamma / ncf

Table 3: workplace breakdown distributions

100 cf. Gutenschwager, K. et al. (2017), p. 137; Ostheimer, B.; Schickert, A. C. (2013), p. 5.
101 cf. Banks, J. et al. (2013), p. 521 p.
102 cf. Ghaleb, M. et al. (2020), p. 13; Gustavo, B. et al. (2019), p. 1024; Gutenschwager, K. et al. (2017), p. 137 p.

As mentioned in Table 3, the majority of workplace breakdowns are realised as stochastic distributions. For four workplaces, no valid distribution could be found based on the available data of the last three years. In their cases the calculated deterministic KPIs mean-time-to-repair (MTTR) and mean-time-between-failure (MTBF) are used. For the manual deburring workplaces, no downtime or maintenance data is available, thus they are not included in downtime consideration.

Due to the varying distribution statements found in literature, a data fitting algorithm is applied to TTR and TBF times. As rejection test, it evaluates which of defined distributions must be rejected because the resulting p-value (explained in chapter 2.4.4) is in the rejection area. The test applies the likelihood function to find the best fitting distribution to each dataset by varying distribution parameters and looking for the maximum likelihood possible. Furthermore, a Kolmogorov-Smirnov test is included as goodness-of-fit test (*German 'Anpassungstest'*), which evaluates the independence of data regarding the considered distribution. As a rejection test, the null-hypothesis regarding independence, meaning that a dataset is not related to a distribution, must be rejected if the p-value is smaller than 0.1 and can be accepted if bigger than 0.1.[103]

The findings are, that exponential distribution can be confirmed for all TBF times, which cannot be reported for TTR times. Thus, exponential distribution is selected if the p-value is in the acceptance area of >0.1. Otherwise, the best fitting distribution is chosen and used in the simulation model. Included distributions are exponentiated Weibull, gamma, alpha, non-central t-distribution and non-central f-distribution. The unique characteristics of these distributions won't be further discussed in this thesis.

As example for the mainly used exponential distribution, the result plot of the rejection test is shown in Figure 13. This graphical representation shows the **probability density function (PDF)**, which describes the distributed probability over all data values. The **cumulative distribution function (CDF)** states the probability that a random variable is less than or equal to a certain value. The **probability plot** is another goodness-of-fit test, that shows how likely the analysed dataset follows the considered distribution, graphically spoken: the closer the data values (red dots) are located to the blue line (distribution), the better the data set fits to the compared distribution.[104]

Figure 13 shows very clearly the big deviation between distributed stochastic TBF durations and a deterministic MTBF KPI. To make this clearly visible, 20 random numbers of the plotted exponential distribution are: [804.83, 126.36, 614.43, 591.71, 340.21, 44.13, 197.45, 449.01, 131.13, 286.14, 78.64, 457.67062981, 94.67, 64.74, 16.40, 201.46, 130.84, 430.60, 44.78, 106.17].

The single calculated MTBF value for 18 months would be 254 hours. Taking that deterministic value in DES would result in big deviation from the reality! This can be stated because a dynamic system behaves differently experiencing long and short durations stochastically instead of always the same deterministic duration.

[103] cf. Law, A. M. (2013), p. 344 pp.
[104] cf. Molin, S. (2019), p. 286; Montgomery, D. C. (2009), p. 95; Law, A. M. (2013), p. 344 p.

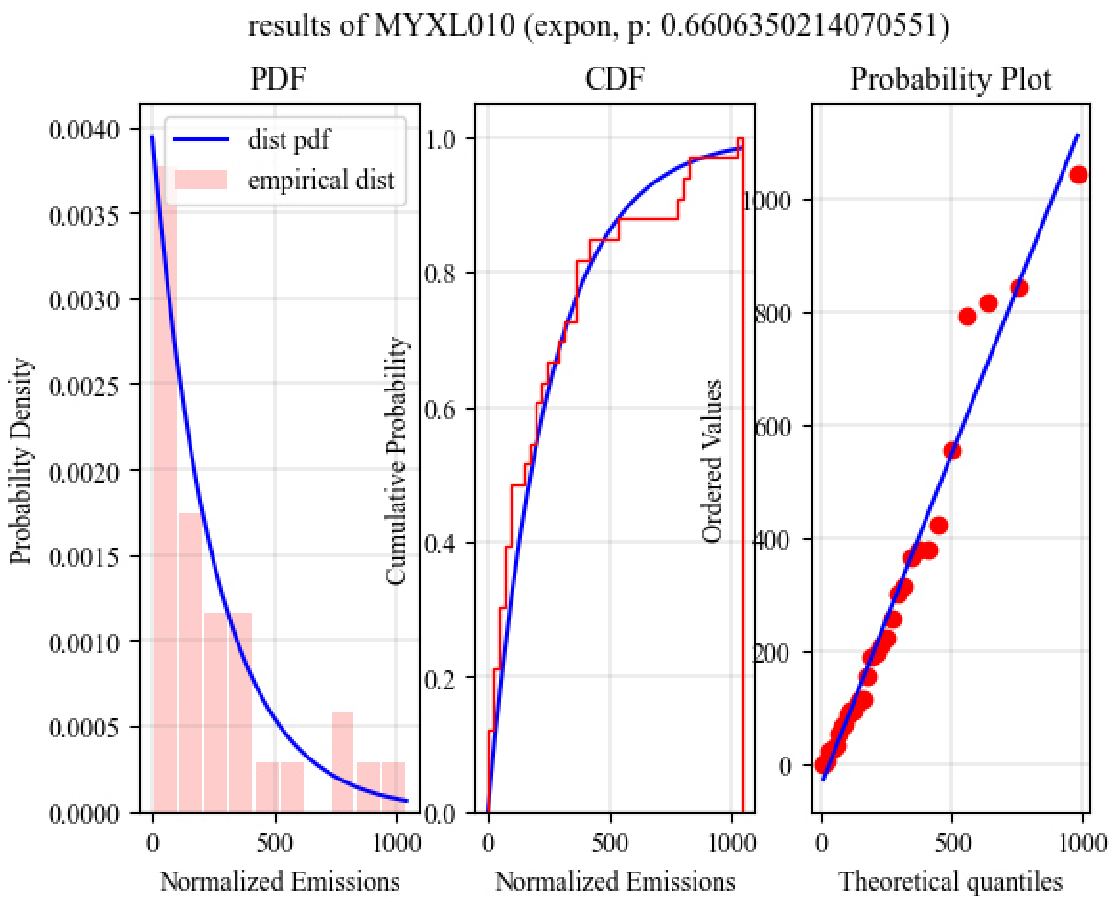

Figure 13: fitted TBF distribution example

An even better example states the TTR times for the same workplace. Between January and June 2022, the workplace had the following breakdown durations in hours: 0.67, 1, 1, 0.5, 1.5, 1, 3, 1.5, 1, 0.5, 0.5, 1, 2.5, 4.25, 0.5, 1, 140. If no distribution was calculated, the MTTR would result in 9.5 hours, clearly a huge gap between the 'real' average downtime.

The determined distributions including their parameters are written into an Excel-file, imported and prepared as Parquet file ready to be loaded from the DES model. If no fitting distribution has been found (relevant for 3 of 43 workplaces) the deterministic values of MTTR/MTBF are used. The four manual workplaces for deburring get excluded as there is no documented breakdown available or existing.

Besides stochastic unplanned downtime, additionally planned **maintenance** times are included in the model by defining a certain number of monthly hours according to the maintenance schedule.

All in all, this chapter gives an overview between the difference of complex preparation for stochastic model elements following a defined distribution versus easy applicable deterministic values, like MTTR/MTBF and monthly maintenance hours which represent static values. Moreover, the importance to be aware of downtime distributions and avoid the usage of mean values for DES is mentioned! MTTR and MTBF are easy KPIs to track the performance development of a maintenance department on a monthly basis for example, but are inappropriate to apply in a realistic and dynamic DES.

3.3.5 Additional Data Preparations

In addition to downtimes and maintenance, there are further influence factors that limit the output of a job-shop, such as sick operators, public holidays or weekends.

Those factors are taken into account as inputs defined in the metadata txt-file. A sick leave factor is determined based on a local report that documents the presence hours of operators, as well as holiday and actual sick leave. A mean value from January to June 2022 is calculated which combined absence time due to holiday and sick leave. This time factor reduces planned shift time in the model, as it results as missing capacity. Public holidays are defined as another simulation input, no operator will be available on those days for work.

Last but not least some more simulation parameters have to be prepared. That involves the shift start, which defines the daily start of the first shift. Additionally, the amount of working hours per shift have to be determined because shifts are not executed evenly for 8 hours in every manufacturing site. Furthermore, a possibility is realised for all importing files to assign column numbers regarding the needed data entities. That allows to import input data generically without a fixed column sequence.

Summing up data collection and preparation

Ensuring V&V regarding data is a very crucial task, which is seeked to fulfil with highest effort possible. Due to the big amount of input data, the phase data preparation results as one of the most time-consuming tasks of the whole model design.

In the first place, raw data V&V measures are applied in form of plausibility checks. All raw data is inspected, unreasonably high values get questioned and excluded, wrong operations are discovered, for example included maintenance jobs instead of operations, and wrong units get transformed into needed formats.

	orders1								orders2							
	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
Material																
	44.0	8.363636	2.412091	4.0	8.00	8.0	8.00	12.0	66.0	9.090909	5.766554	4.0	4.0	8.0	12.00	24.0
	22.0	8.363636	2.440637	4.0	8.00	8.0	8.00	12.0	32.0	7.656250	4.519559	2.0	4.0	8.0	8.00	24.0
	22.0	2.090909	0.610159	1.0	2.00	2.0	2.00	3.0	26.0	2.307692	1.568929	1.0	1.0	2.0	2.75	6.0
	22.0	4.181818	1.220319	2.0	4.00	4.0	4.00	6.0	36.0	4.416667	2.589815	2.0	2.0	4.0	6.00	12.0
	44.0	2.090909	0.603023	1.0	2.00	2.0	2.00	3.0	67.0	2.253731	1.105678	1.0	2.0	2.0	3.00	5.0
	44.0	4.090909	1.177664	2.0	4.00	4.0	4.00	6.0	75.0	3.786667	2.176616	1.0	2.0	4.0	4.00	10.0
	66.0	4.090909	1.173126	2.0	4.00	4.0	4.00	6.0	98.0	3.530612	2.164398	1.0	2.0	3.0	4.75	10.0
	20.0	1.400000	0.502625	1.0	1.00	1.0	2.00	2.0	36.0	1.750000	1.024695	1.0	1.0	1.0	2.00	4.0
	150.0	2.880000	1.698479	1.0	1.00	3.0	4.00	8.0	66.0	3.621212	2.429088	1.0	1.0	3.0	5.00	9.0
	150.0	2.880000	1.698479	1.0	1.00	3.0	4.00	8.0	56.0	3.803571	2.384950	1.0	2.0	3.5	5.00	9.0
	100.0	2.880000	1.701336	1.0	1.00	3.0	4.00	8.0	40.0	4.000000	2.480695	1.0	2.0	4.0	5.00	9.0
	159.0	25.264151	21.600941	1.0	6.00	12.0	45.00	88.0	319.0	15.448276	8.447190	1.0	10.0	15.0	20.00	38.0
	212.0	24.981132	21.689310	1.0	6.00	10.0	45.00	88.0	375.0	17.090667	9.122007	1.0	11.0	15.0	21.00	41.0
	150.0	2.880000	1.698479	1.0	1.00	3.0	4.00	8.0	67.0	4.000000	2.552479	1.0	2.0	4.0	5.00	9.0

Figure 14: pandas example print of grouped DataFrame in Jupyter Notebooks

Since the final model input files are saved as binary Parquet files, Jupyter Notebooks is used regularly to inspect results of prepared data. Pandas offers a wide range of mathematical functions and groupby options to analyse aggregated data efficiently, like shown in Figure 14. Pandas and Jupyter are further explained in the upcoming chapter.

Following the V&V procedure model, the results raw and prepared data are successfully concluded. Thus, the created Parquet input files are ready to be imported from the executable model.

3.4 Formal Model

Based on the knowledge gained in the system analysis phase by attending a local workshop and conducting interviews with local colleagues, first steps are undertaken to formulate a formal model. One goal of this thesis is to create a generic framework that allows to model various discrete systems. Thus, several decisions are necessary regarding choosing an appropriate data analysis tool, computed data storage file format, a generic input possibility for simulation users and lastly a suitable simulation tool.

Since the core element of this study should be a valid simulation model, all additional tools should be compatible with the finally used **simulation tool**. After some research, the **Python library 'salabim'** is chosen based on a couple of reasons. First, Python is a very popular programming language in data science and machine learning with a large variety of open-source libraries to use on top or besides the actual simulation. Secondly, salabim offers a comprehensive online documentation including modelling examples and an active Google group can be joined where regularly questions are discussed and answered, mainly by the core developer Ruud van der Ham himself. SimPy is probably the most popular DES library in Python, but salabim has its own developed event scheduler and supports additional model concepts that allow to activate, passivate and hold processes and entities enabling the process-interaction approach, mentioned in chapter 2.4.1. Moreover, the activity-scanning approach can be implemented by salabim's component feature 'State' as well.[105]

As data analysis tool, **pandas** is chosen because it is a Python open-source library as well as it is very popular in the data science community. It is built on top of the NumPy library, enabling very fast and efficient mathematical operations on single-type arrays. Those arrays are called DataFrames in pandas, which are like data tables. Additionally, pandas provides wrappers around the matplotlib library which enables a wide range of plotting functionalities. Wrapper functions use the capabilities of different libraries and offer them in a simpler interface for repeating the same functionality. This principle is called abstraction and is a common method in object-oriented programming.[106]

A handy additional tool used for small and fast data examinations is **Jupyter Notebooks**, as integrated development environment (IDE) giving the opportunity to get quick insight into computed data or code snippets. In Figure 15, an example of workplace raw data using the pandas info() method is shown, screenshotted out of a Jupyter Notebook. This method gives a short overview about existing columns and their data type as also the

[105] cf. Lang, S. et al. (2021), p. 983 pp.; van der Ham, R. (2021).
[106] cf. Molin, S. (2019), p. 43 p.

number of non-null values of each column. In general, PyCharm is used as IDE running Python 3.9.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 55 entries, 0 to 54
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Objekt-ID           55 non-null     int64
 1   Equipment-ID        39 non-null     float64
 2   Arbeitsplatz        55 non-null     object
 3   Werk                55 non-null     int64
 4   Bezeichnung         55 non-null     object
 5   Hierarchie          55 non-null     object
 6   groupName           55 non-null     object
 7   Schichten           46 non-null     float64
 8   AnzahlAPLs          46 non-null     float64
 9   Mitarbeiterfaktor   46 non-null     float64
 10  Leistungsgrad       55 non-null     float64
 11  Nutzungsgrad-Pers   55 non-null     float64
 12  Warteschlange_1     43 non-null     object
 13  Warteschlange_2     14 non-null     object
 14  Kommentar           7 non-null      object
dtypes: float64(6), int64(2), object(7)
```

Figure 15: pandas info() method

Having the simulation and data analysis tools defined, Parquet was chosen as data type to store computed raw input files, due to the good compression factor and compatibility with pandas. Parquet is a binary file type, meaning that it’s not human-readable and needs software to interpret the stored data. It allows to save information with a specified data type such as numeric (int64, float or double), boolean or byte array which can be used for text or objects. Data is stored in columns instead of rows and it supports the possibility to natively compress data within its files efficiently.[107]

As generic simulation input possibility, a .txt-file is created, which gets filled with all parameters needed to perform a simulation run, such as data path, file names, simulation start- and end time, units, column names or specific model parameters, listed in chapter 3.5.1.

After the first data preparations, described in the previous chapter 3.3, a process diagram is created which visualises the main input files and process steps, illustrated in Figure 16. Starting with the yellow input files (orders, workplaces and work in process (WIP)), which are realised as DataFrames (df) in the model, the simulation parameters are loaded from the txt-input file, into the simulation environment. This data gets combined and associated with the model’s entities (coloured in white) including a respective plant where all workplaces exist and orders should get processed. A production planner then formulates production orders and releases them accordingly to a planned start date to the shopfloor by placing it to the first relevant queue.

After a linked workplace has free capacity for a next operation of a production order, there might be some optional setup time, followed by a defined processing time. After an operation is finished, feedback will be given to the production planner and the related production order. The production order is then waiting in the output queue of the handled

[107] Inamdar, A. (26.09.2020).

workplace. From there a transporter will transport it to the next workplace defined in the working plan or to the finished goods warehouse. Like shown in the diagram, orders will be released daily and discrete simulation steps are implemented in seconds. Every simulation run will be evaluated regarding the mentioned KPIs (highlighted in orange), such as WIP, daily orders, queue lengths and waiting times or utilisation, occupancy rate and availability of workplaces. Those KPIs will be explained in more detail in chapter 3.5.

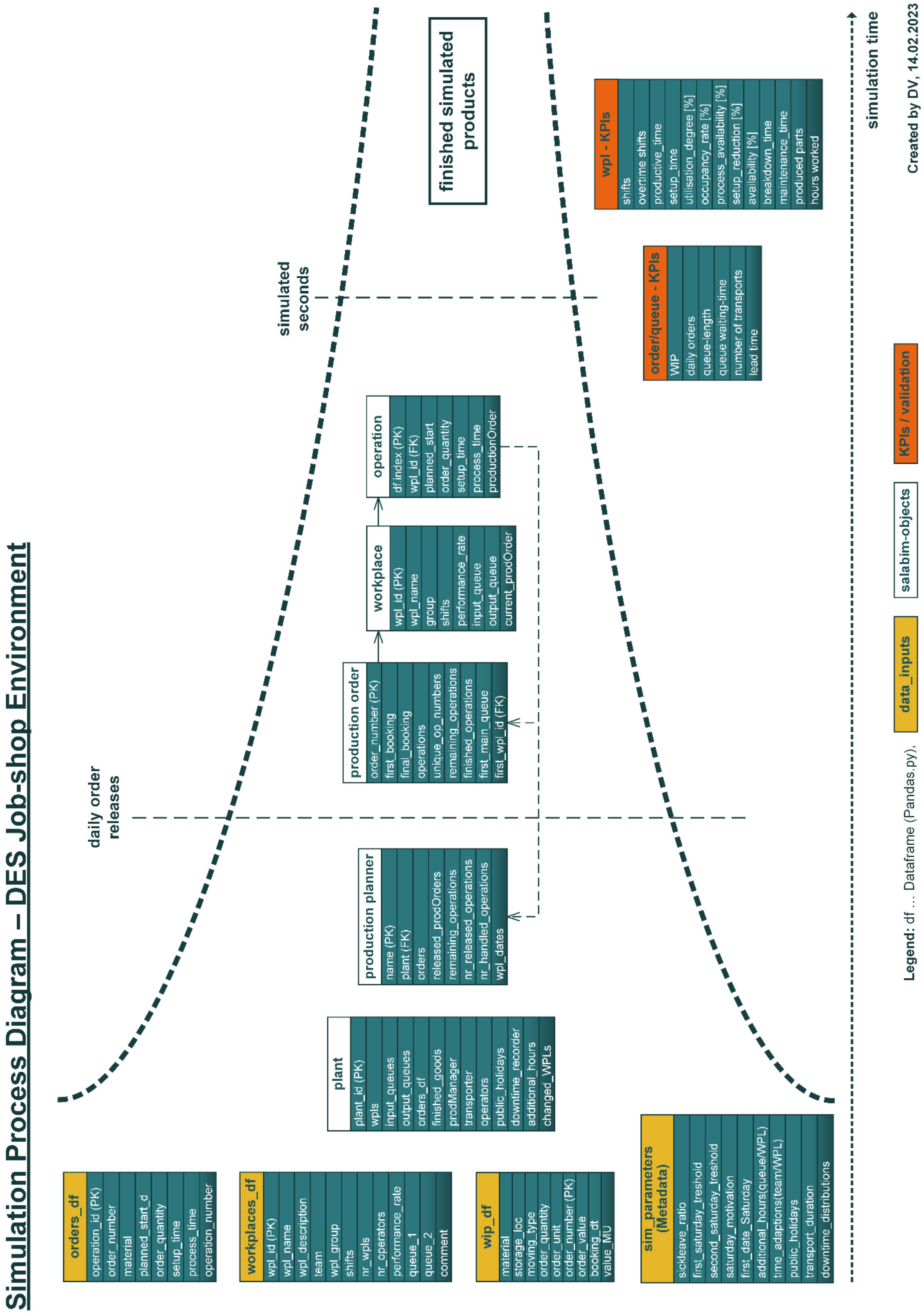

Figure 16: simulation process diagram

In order to design a valid model that represents the reality of the complex handled production system with over 40 workplaces and covering 9 different production technologies, the following simplifications have been made:

- The ERP system's working plans do not have alternative workplaces included, thus the machine technologies have to be combined into common queues. Meaning that all lasers for example can work on the same orders. Restrictions such as laser cutting power, press brake force or allowed part-dimension are not taken into account. Distinctive technologies like autogenous cutting, special milling, drilling or straightening workplaces are discussed with local process engineers and handled separately as accurate as possible.
- WIP only includes production orders with at least 2 operations. Based on the given data situation, only operation confirmation timestamps are available (retrograde booking), thus no WIP and lead time can be determined for production orders with single operations (e.g. only laser cutting). This case applies to about a third of all orders. Additionally, a small number of orders (less than 1% in total) have to be excluded because they are processed at additional workplaces (like welding) between the considered workplaces, as so-called net orders. These limitations are only relevant for the KPIs WIP and lead time. Nevertheless, all orders are simulated.
- The amount of WIP value only includes the raw material value, meaning that no additional added value is taken into account.
- 2% of material numbers handled in future scenarios have not been known in 2022 because they got renamed or were newly designed because of new models. Their value is not included in the WIP consideration.
- No transport matrix is created (e.g. by defining distances between workplaces and calculate the needed time based on methods time measurement (MTM)). The model includes one transporter, which transports all production orders to the next workplace based on a small duration as defined model parameter.
- This model assumes 100% raw material availability and does not consider internal transport as a potential bottleneck of the production system.
- Downtimes are mainly considered stochastically, like shown in Table 3 and described in chapter 3.3.4, maintenance is covered by deterministic monthly hours according to information from the maintenance department. Nevertheless, an operation does not get interrupted, but the model checks if downtime or maintenance is 'necessary' before a new operation is started.

The formal model can be summarised as illustrated process diagram in Figure 16, which builds the basis for the executable model. So far, a data analysis tool, computed data storage file format, generic input possibility for simulation parameters and a simulation tool are chosen and represent the basis to create a DES model. The 9 V&V criteria completeness, consistency, accuracy, currency, applicability, plausibility, clarity, feasibility and accessibility are considered constantly and lead to the listed simplifications above.

3.5 Executable Model

Having ensured that all previous phases are conducted in a verified and validated manner by applying constant V&V measures, the core element for future statements and conclusions can be setup as executable model.

Like discussed in chapter 3.4, the **Python library salabim** is chosen as **simulation tool** for this thesis because of its versatile application possibilities and wide enrichment options through running in Python. Moreover, the option to freely choose between event-scheduling, activity-scanning and process-interaction approach, described in chapter 2.4.1, including statistical tracing functionalities are big benefits of salabim.[108]

Since it is a level 1 simulation tool, defined in chapter 2.4.3, the modeler has to be familiar with Python as the creation of a model is only possible by writing Python code. Compared to a commercial DES tool, like Arena or Tecnomatix Plant Simulation for example, salabim does not offer a graphical user interface, although animation of a created model is possible. Nevertheless, this model will not be animated. Therefore, the validation focus is on different methods and conclusions regarding the simulation goal want be achieved without an animation. As assistance the software allows different powerful evaluation features for V&V tracing every event in the Python output console or using the inbuilt statistical reports about queue utilisation and state.[109]

[108] cf. van der Ham, R. (2018).
[109] cf. Lang, S. et al. (2021), p. 983.

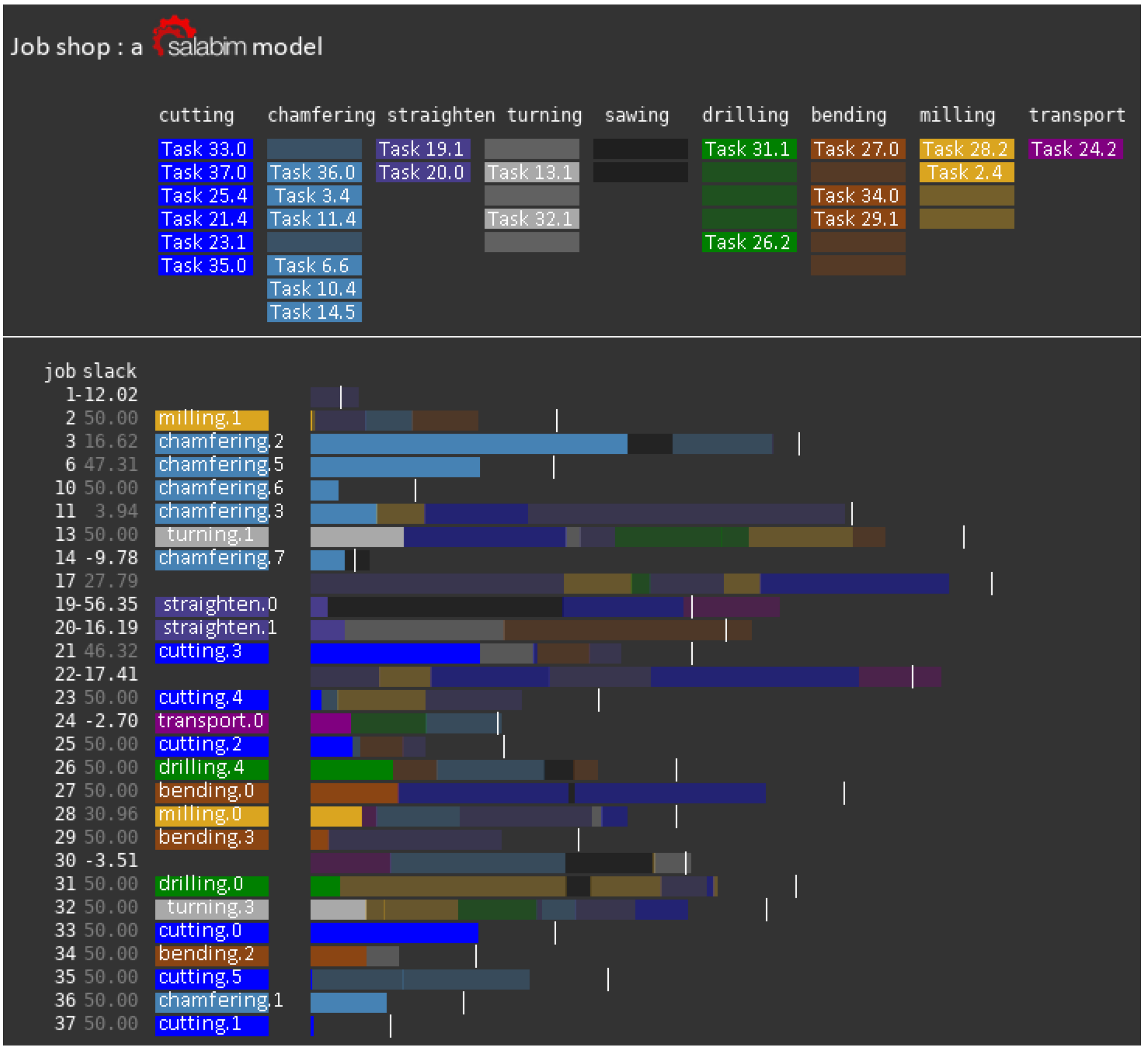

Figure 17: salabim animation example[110]

For better DES insight, an animation example is presented in Figure 17, which does not represent the created model, but similar handled technologies and amount of workplaces. The sequence of orders does not represent the considered environment, but it can be noticed that for example all cutting workplaces are occupied in the displayed moment with different operations. Apart from that example, the jobs 17 and 22 have to wait in the queue of straightening because both workplaces are already busy. Note that in this given example, only in exceptional cases transport is necessary after an operation is finished. The source code of this animation example is provided in salabim's Google group and has been adopted regarded used workplace names.

In this phase, the complete simulation framework will be realised and set up. The base components and classes plant, production planner, production order, workplace and operation were already mentioned in chapter 3.4. Additionally, a transporter is implemented which moves processed material from one workplace to the next one or to the finished goods warehouse. As stated in the listed limitations in chapter 3.4, no variable transport time resulting from different distances between workplaces is implemented and can be mentioned as improvement for the future. As a consequence,

[110] cf. Ruud van der Ham (2020).

by accepting this limitation it is assumed that transport is no potential bottleneck of the examined system.

Before explaining the handled approach of every simulation run in chapter 3.5.1, some more simulation parameters have to be defined. The following stated setting parameters have big influence for the result of a single simulation run and are entered in the txt-input file. The **simulation duration** defines start and end date of each simulation run. Instead of defining dates it could also be realised as duration in seconds, minutes or hours. The user can decide that the model should **skip multiple setup**, which means that setup won't be executed if the same material has been processed before the current material at the workplace. Additionally, it is possible to **deactivate downtimes** in general, doing so no breakdowns will be simulated. As this framework initialises the model with 0 amounts of production orders in queues and develops from a transient phase into a steady state, a **validation period** should be set to evaluate validation measurements only for a respective time span. In addition, the user can decide if WIP is evaluated on a daily or weekly basis, called **WIP time interval** in the txt-file. Another parameter is called **first queue length threshold**, which enables the possibility of overtime shifts on Saturdays. If this selected amount of production orders is exceeded in a queue on Friday evening, it triggers one additional shift per week, but logically only possible for workplaces with one, two or three shifts. The **second queue length threshold** acts similar to the first queue length threshold, additionally adding a second shift on a Saturday if the queue exceeds a second number of production orders. The **Saturdays operator motivation** defines a percentage how many operators are willed to additionally work on a Saturday. If for example 0.5 is set, 50% (rounded up) of all operators will be available for one or two shifts on Saturdays, according to the set first/second queue threshold. To rise the accuracy of used overtime shifts, the optional **first date Saturday** parameter defines a date when the first overtime shift is possible to be applied. Moreover, the restriction **maximum allowed hours of overtime** is implemented to avoid too many overtime shifts. By defining a number of hours per team or workplace, it is possible to restrict the amount of overtime that is 'allowed'. Last but not least, the parameter set **time adaption** is implemented due to various reasons, such as no defined alternative workplaces in the ERP system or difficult determination of standard times for all processed materials. This feature adapts the amount of setup or process time of a workplace or complete queue. Otherwise, huge queues would result for specific workplaces and would influence the whole system. This parameter is further explained in chapter 3.5.2. Affected workplaces are discussed in detail with the industry partner.

In order to comply with reality, physical **operators** are realised in the model who need to be present at a workplace before an operation can be processed. Here the activity-scanning approach is used, constantly checking if an operator is present, otherwise the workplace must wait until an operator shows up again before starting the next production order. This means, that workplaces themselves do not know shift times, attending operators (who know shift times, like in the real world), determine if a workplace can process operations or not.

Another conducted reality feature is about a ruleset that allows alternative workplaces by two different approaches. Firstly, in the workplace input file a technology queue can be specified. That means, that production orders assigned to one workplace can be handled from all workplaces of the same **technology queue**. This method will further be called **bucket-queue**, as orders for the same workplace technology are accumulated in buckets. In one bucket also queue-rules like FiFo are valid, as soon as one bucket workplace has free capacity, the production order with the longest waiting time in the queue will be processed next. This method enables for example to summarise all chamfering workplaces in one bucket, but allows to consider the autogenous cutting workplace in a separate queue instead of adding it to the laser bucket.

Secondly, it is possible to define **up to two unique queues for one workplace**. A second queue-definition acts like an alternative resource by assisting another one. As an example, a specialised drilling workplace can provide capacity to production orders that are referenced to a more general drilling workplace, but not the other way around. As soon as the special drill has no orders in its own queue, it will process orders from the general drill. If no unique queue is defined in the workplace input file, the WPL is automatically added to its WPL-group bucket-queue.

During the conceptual model phase, it turned out that during January and June 2022 three **new workplaces** were added to the system and started operation. To cover such cases, the comment-column of the workplace input-file is used by adding new_YYYY-mm-dd the correct operation start date. As soon, as the simulation date matches that defined date, the production planner will activate the respective workplace, which will then offer additional capacity to the assigned queue. This feature is used for the added laser, bending and chamfering machines that were added in March and April 2022.

In general, the complete listing of all involved classes of this simulation framework including their relations can be found as UML-diagram in Figure 18. The upper left section is about the import process of input files and saving process to Parquet-files. The class TXTReader shows the different components and parameters of the txt-input file (‘Metadata’). The main() class starts a simulation which will be explained in the following chapter 3.5.1. Additionally, the class simValidate is all about validation, evaluation and analysis of the current performed simulation run by printing defined measures into the output console, exporting result reports and generating diagrams about WIP, lead time and waiting time boxplots, like shown in Figure 21, Figure 22 and Figure 24.

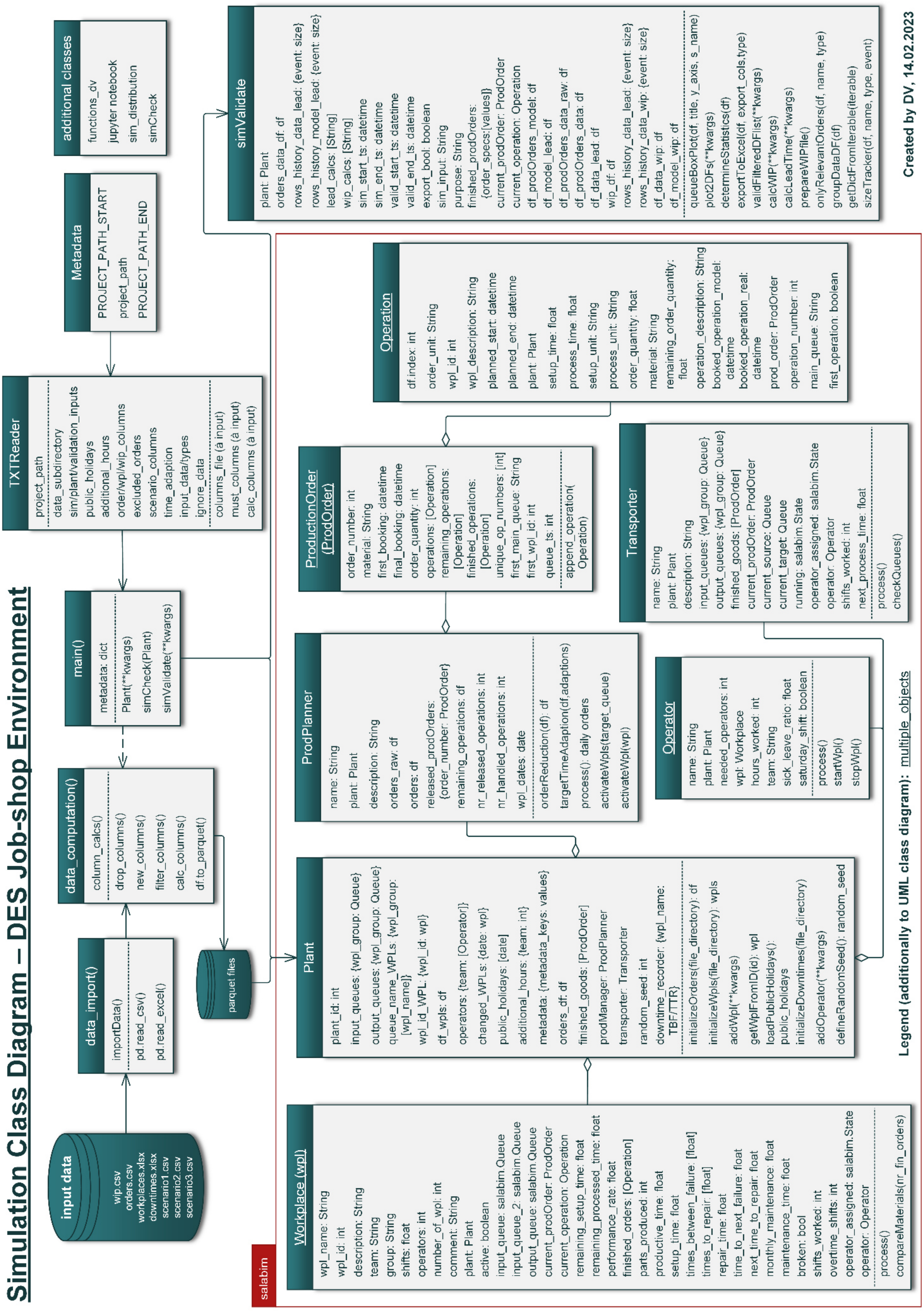

Figure 18: simulation class diagram (UML)

3.5.1 Model Sequence Logic

This subchapter describes the performing sequences of every simulation run. As requirement, all steps of chapter 3.3 are fulfilled and complete data is ready for import and available as Parquet files.

When starting the simulation via the main() method shown in Figure 18, first of all the analysed plant is initialised, since the handled job-shop is located in one single plant, only one is needed, but theoretically multiple plants could be created. By that initialisation a couple of additional initialisations take place, that include:

1. Breakdown distributions are generated according to defined distributions.
2. All workplaces with defined shifts get created. If a workplace is not productive from begin on, 'new_date' is defined in the comment-column of the input file in order not to get activated from start. The generated TBF, TTR and maintenance times from the previous step are referenced to the corresponding workplace. Additionally, for every workplace an operator is created.
3. Defined queues get initialised and referenced to the correct workplace. As mentioned in the end of chapter 3.5, it is possible to assign up to two queues per workplace.
4. Afterwards input operations get imported and loaded into a DataFrame, including transforming the imported time units to the used simulation time unit.
5. Then a so-called production manager is created who is responsible for production order aggregation and release. Orders are released on a daily basis to the shopfloor, which means that operations with the same production order number get combined into one object and placed into the queue of the first operation's workplace. As release date, the planned start date of the ERP system is taken. Further it is the production planner's task to adapt setup or process times mentioned in chapter 3.5. He additionally checks daily, if one of the inactivated workplaces can be activated.
6. For transport reasons from an output-queue to the next queue (either a workplace or finished goods) at least one transporter is needed and initialised. As this thesis is focussing on workplace bottlenecks and assumes 100% material availability, inefficiencies in transport are neglected.

After the initialisation of all model components, a daily procedure is repeated until the simulation time is finished. This scheme always starts by daily production order releases of the production planner and processing workplaces. Those firstly check their main queue and secondly their second queue if production orders are available to be processed, as soon as an operator is present. This model only includes FiFo as prioritisation rule, meaning that always the order that has been waiting the longest in the queue will be taken first.

A workplace can only process operations if an operator is assigned. This setting is implemented by the activity-scanning approach and salabim's state method. Operators know about shift times and get active and passive according to the defined shifts of a workplace. If no production order is in a related queue, the workplace passivates and will be activated by the transporter as soon as a production order is dropped in its input queue. After every finished production order, the workplace checks if it can continue with processing or if it has to passivate because of breakdown or maintenance.

As soon as all operations of a production order are finished, the last step is that the transporter moves it from the last output queue to the finished goods queue, which represents the end of the considered model boundary.

In order to ensure a close correspondence between the created model and the reality job-shop, big effort has been applied to add as many reality aspects to the model as possible. Regarding achieving a verified model, the most critical part represents the allocation of workplaces and their respective queues. This has required a constant exchange with process experts of the industry partner.

3.5.2 Model Validation

As mentioned in chapter 2.4.3, it is essential to apply V&V measures constantly when setting up a computerised model. Thus, at the very beginning of the model's setup, several methods are implemented that keep track about V&V. This involves the correct occupancy duration of workplaces to perform setup and processing, the number of handled operations and production orders, correct sequencing of orders from one workplace to another and the right implementation of defined shifts for example.

Moreover, as mentioned in chapter 2.4.3, operational validation determines if the output of an executable computerised model has sufficient accuracy. To approve this statement, a couple of measures and techniques are carried out, all of them are theoretically explained in chapter 2.4.4.

Performed test runs at the beginning of the model design phase with short simulation time periods, analysing every simulation step and validating the operation time of an operation according to the original setup and process time. This method represents a **detailed trace analysis**. Small time adjustments like time adaptions, excluding workplaces, varying shift times, performance levels, different overtime motivation values and changing maximum breakdown times validated the model regarding **extreme-condition testing**. As additional checks, **sensitivity tests** are conducted by modifying the sickness ratio which approves its correct implementation. **Internal validity test** is executed to approve the chosen distributions for all stochastic workplace breakdowns. To avoid the case, that unusual overtime shifts have to be conducted by similar workplaces, a maximum repair time parameter is implemented with a limit of 100 hours. Another realised **trace analysis** compares the real stamped presence time hours of every team to the virtual total operator presence time.

The **core validation method** is based on the key performance indicator **WIP**, which is measured in monetary units (MU) resulting from production orders waiting in queues or occupying a workplace for setup or processing. This metric gets compared between reality bookings and model results. System boundaries are the first and final booking of a related workplace, meaning that a production order is added to WIP as soon as the first operation is finished and removed as soon as it is finished at the last workplace. This WIP comparison calculation can be defined on a daily or weekly basis via the txt-input file. The shown KPI diagrams in chapter 3.6 result based on a delta evaluation between all WIP that entered minus all WIP that left the considered production environment on a daily basis.

Besides the actual WIP, also the number of production orders is compared, based on the same rules as WIP in monetary units.

According to the booking timestamps, it is possible to state in which time period which orders have been able to be finished in reality. A big influence factor for the overall simulation of a production system and the validity of the model are correctly defined setup and processing times. Those also-called standard or target times are normally already determined and deposited in the ERP system in order to perform capacity planning. The definition of those times is a core task of Industrial Engineering and a challenging exercise due to the conflict of interests between balanced operator workload and maximum possible company output. Nevertheless, those times are essentially needed to consider correctly how long a workplace is blocked by any operation, heavily influencing the queue dynamics of each workplace or group, where production orders steadily arrive from predecessors.

As the determination process is not a trivial task, not 100% correct target times are a possible issue with big influence on the behaviour of a model. Besides too high defined target times, two other possibilities are identified that unrealistic simulation results, like shown in Figure 19, occur. Firstly, a workplace can be staffed by far more than originally planned in the shift plan which would result in overtime hours in reality. Secondly, material is processed at non-bottleneck workplaces which are not defined in the ERP system's working plan. Those two reasons are only possible to a minor grade, because the total sum of reality presence hours match to the simulated working hours.

Data uncertainties are a common problem in simulation in general. In order to formulate a valid model that represents the observed job-shop, **adaptation parameters** are implemented that allow the modification of pre-defined standard times:

1. The local process experts have already evaluated a performance rate in the past, where they compare the sum of standard times of produced orders per month with the sum of operators' presence times. As a result, a utilisation degree (also called performance or productivity rate) is known, which gives information about the performance or correctness of the existing standard times. For workplaces where a big performance rate has been determined, this factor is set as variable in the workplace input file. All affected target times are modified by that factor, which is done directly for every process of a workplace.
2. When executing the model, some queues show unrealistic long waiting time, resulting from too big standard times, unregistered overtime or not defined alternative workplaces. This can be proven because a certain amount of production orders was actually handled and finished in reality, but the model cannot finish them in time. Thus, a second standard time adaption option is implemented which is carried out during the production planner's initialisation. Based on a defined factor in the txt-input file, the possibility is realised to define whole teams or workplaces where the setup or process time gets adapted. See Figure 19 as example for 2022, how simulated WIP levels (yellow) develop if this feature has not been implemented. This shows the result of unrealistic queues with mean waiting times of several hundred hours.

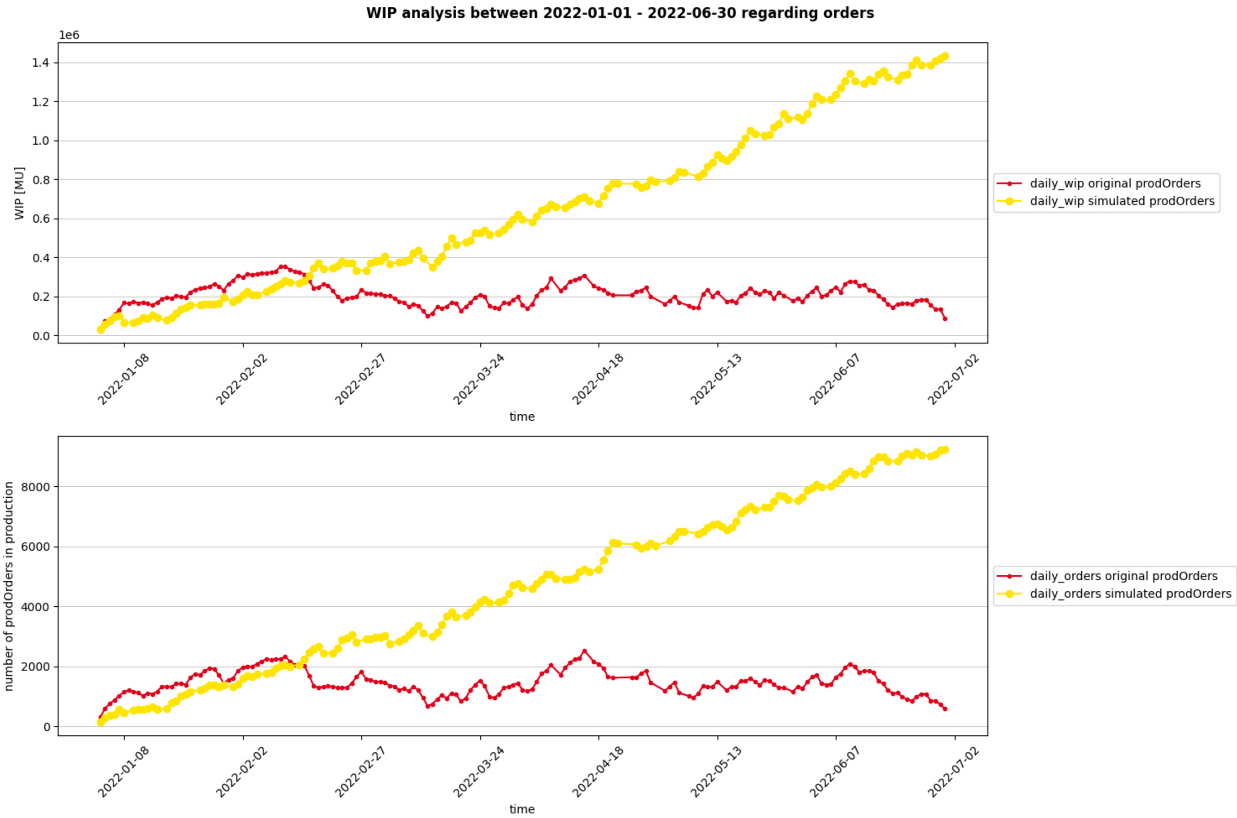

Figure 19: exceeding WIP levels

As soon as the critical workplaces and their resulting queues are identified and adjusted, a more reasonably WIP development results. The applied changes are discussed below Figure 22.

The simulated WIP value shown in yellow in Figure 19, is measured in monetary units representing the raw material price, thus no value-adding value is added or considered. In Figure 21 the **transient phase** at the beginning of the simulation time can be identified clearly until the beginning of March. Page and Kreutzer describe different techniques to detect a valid **stationary phase**. Besides graphical methods also statistical methods are listed like the **Crossing the Mean** approach, plotted in Figure 20. This technique determines the mean value of the reality system and indicates a stationary phase as soon as the simulated values have crossed the mean value three times.[111]

[111] cf. Page, B.; Kreutzer, W. (2005), p. 174 pp.

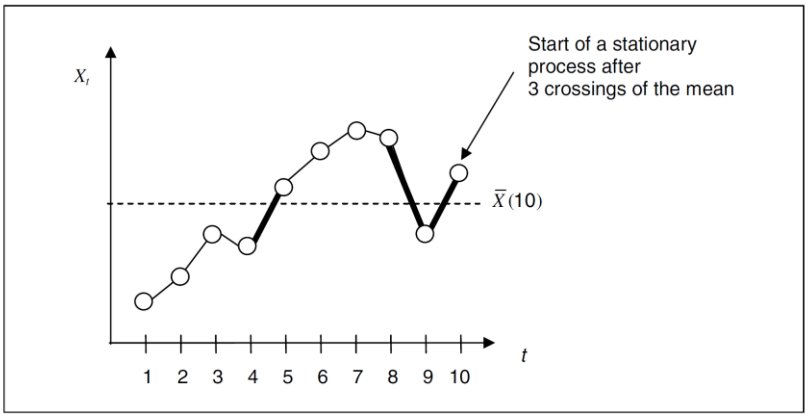

Figure 20: Detecting stationary phase by *Crossing the Mean*[112]

In the case of this thesis the reality's mean value of total WIP is 200.976 MUs. The simulated WIP exceeds that value the third time on the 23rd of March 2022, thus this date is chosen as validation start, marked by the left vertical line in turquoise in Figure 21.

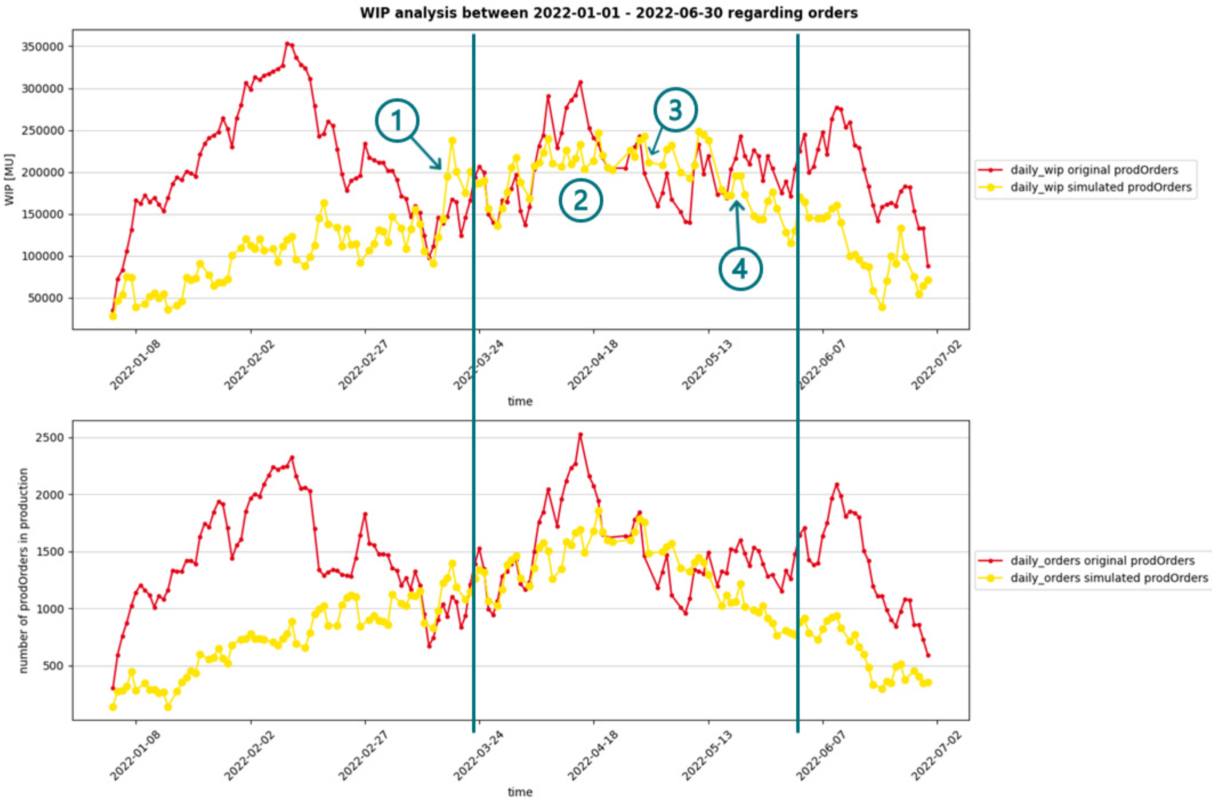

Figure 21: WIP development 2022

The overshoot of WIP between the 16th and 19th of March results because over 2.300 hours of processing time are planned in the ERP system, but only 2100 hours are actually processed. That means that the local management team levelled the workload in reality. A levelling functionality is not part of the model. As a result, WIP overshoots because all planned production orders are released to the shopfloor. June has to be excluded because at the end of the simulation, production orders are not finished, hence they don't

112 Page, B.; Kreutzer, W. (2005), p. 177.

have a final booking date and are not included in the WIP/lead time calculation. Additionally, orders from July were started in reality, so this would result in a mixture. Other mentionable time periods are marked with circled numbers. The area near number (2) represents the week before Easter, where production orders were started earlier which resulted in higher WIP as planned. The period around (3) is characterised by public holidays around the 1st of May, which is difficult to simulate because of varying operator presence. The arrow at (3) points to a higher modelled WIP, compared to reality which is caused by more finished material than planned new orders in reality. Number (4) shows a period where more process time was handled in reality than originally planned. In general, it must be noted, that due to stochastic distributed workplace breakdowns, a 100% same WIP level could only be achieved by a deterministic model. The resulting disadvantage is that such a model cannot be used in a meaningful manner for future analysis compared to a stochastic one.

In addition to WIP measured in MU, also the amount of production orders is considered, which represents the lower graph in Figure 21. Both KPIs are calculated on a daily basis.

The validity regarding WIP is measured by applying tests include the two-dimensional t-test, RMSE, MAE and R-squared, described at the end of subchapter 2.4.4, for every simulation run.

Regarding the applied two-dimensional t-test, as null hypothesis the following statement is chosen: The mean values regarding WIP should be equal $H_O: \mu_1 = \mu_2$, the alternative hypothesis would then be, that they are unequal: $H_A: \mu_1 \neq \mu_2$. As the common two-dimensional t-test demands equal variances of both data sets, Levene's test is applied in order to test the null hypothesis that all input samples are from equal variances. The result is a p-value below 1%, thus the null hypothesis must be rejected, meaning that variances of reality and simulation results vary significantly and are not equal. Therefore, Welsh's test is applied, representing a special variant of the two-dimensional t-test, that allows different variances. This statistical pre-test implies the application validity for Welsh's two-dimensional t-test, handling unequal variances.[113]

The two-dimensional t-test, more specifically Welsh's test, is realised using the Python SciPy library, which returns a p-value as result, that quantifies the probability of observing corresponding mean values, assuming that the null hypothesis must not be rejected. A p-value bigger than a chosen threshold of e.g. 10% indicates, that the simulated results reflect the reality.[114]

All performed simulation runs include a validation part of the mentioned KPIs above, and parameter settings are accepted as soon as Welsh's p-value is greater than 20%. Achieving this result concludes that the null hypothesis of equal mean WIP-values is proven and must not be rejected.

In addition to WIP, a second used metric to evaluate a simulation run is a conducted lead time comparison on a monthly basis. It involves the total time production orders need to get through the job-shop. Similar to the WIP determination, first considered timestamp is

[113] cf. SciPy community (04.01.2023a).
[114] cf. SciPy community (04.01.2023b).

after the first workplace and final of the last one. In Figure 22 it can be seen that the model's mean as also median lead times are close to the reality ones.

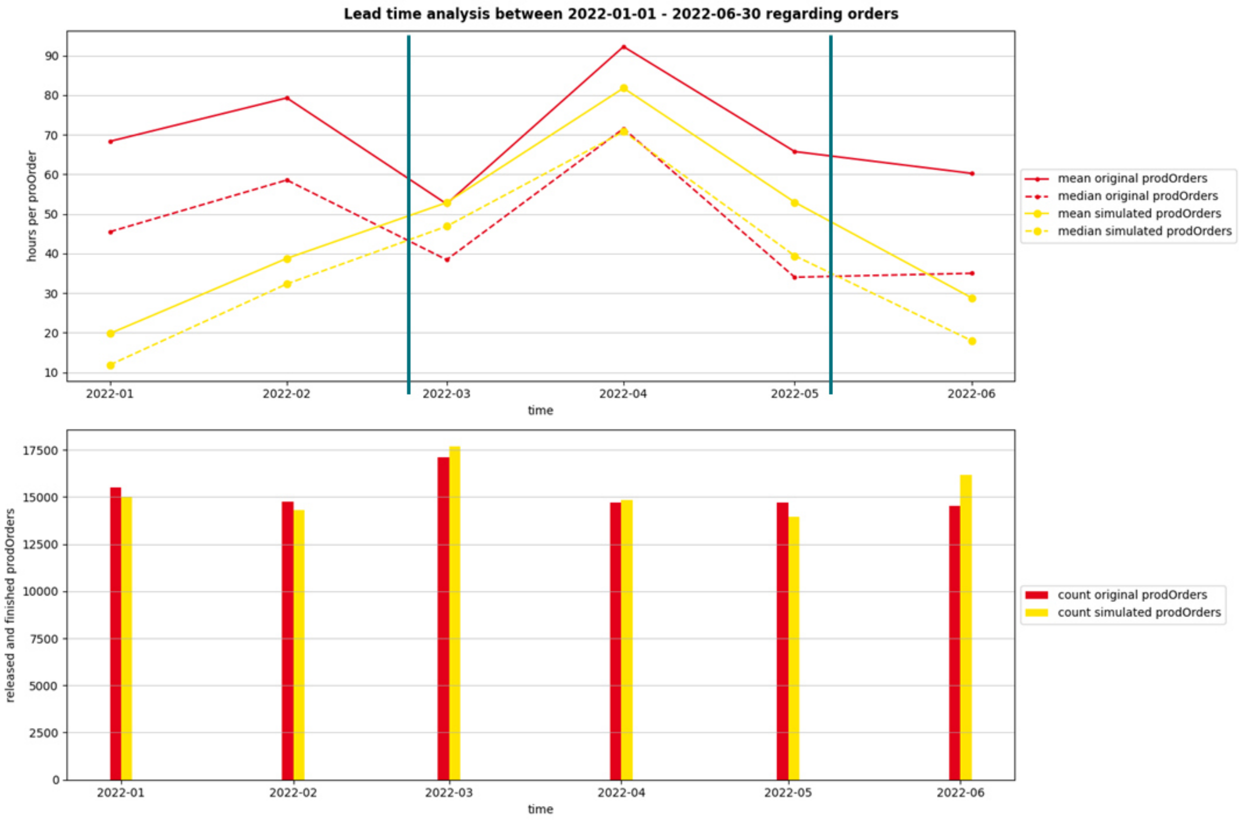

Figure 22: lead time development 2022

Besides the KPIs WIP and lead time, every simulation run creates a couple of reports which are exported as .csv files and include various KPIs regarding workplaces and orders, such as:

- Handling time which includes processing and setup time, worked shifts and amount of overtime shifts.
- Breakdown time from unplanned breakdowns based on defined distributions or MTTR as sum and additionally all occurred TBF and TTR times. Planned maintenance hours are printed as well.
- Based on defined production KPIs in chapter 2.2, the utilisation degree is mentioned in the reports as well.
- In the used MES (Manufacturing Execution System) of the cooperating company, **availability** is calculated as a thoughtful combined metric and thus used. The including KPIs are defined by the VDMA standard and already mentioned in chapter 2.2. Only the setup rate has to be changed to a so-called setup reduction, otherwise it cannot be used:[115]
    - Setup reduction (*German 'Rüstzeitminimierung')*: A measure setting the processing time T_P in context with the defined handling time T_H, which includes setup time. This KPI would be 100% if no time was needed for a workplace change over.

[115] cf. VDMA 66412-1:2009-10, p. 9 pp.

        - $setup\ reduction = \frac{T_P}{T_H} * 100\%$ (3.1)
    - By multiplying the occupancy rate, process availability and setup reduction, a combined KPI results which gives consolidated information about how much a workplace is actually really needed (occupied) and available (breakdown and setup) for value-adding processing time.
        - $availability = occupancy\ rate * process\ availability * setup\ reduction$ (3.2)
- Queue-statistics including mean and median length of each queue, number of entries, as also mean, median and maximum waiting time that production orders had to wait until being handled.

In the end the following KPIs are considered as most interesting to mention regarding simulated bottlenecks for 2022's validated model results:

- Long queue waiting times result for:
    - Lasers and bending. Until the new machines start processing in March/April 2022, many overtime shifts are necessary in the first months between January and March to handle the big workload. Additionally, the setup process of bending machines leaves manual improvement potential for operators, thus setup target times have to be reduced.
    - Straightening, because the original standard times include the case that every part has to be straightened which does not represent the reality. Thus, times have to be reduced as well, otherwise too big queues would result.
    - Milling workplaces, which are deeply discussed with process experts, because they have special properties that complicate alternative workplace definitions. Based on that result the queues for two milling machines (MF020 and MF024) are very high. Therefore, target times have to be reduced as well.
    - All other workplaces do not result as a bottleneck which aligns with the local experience.
- High occupancy rates are clearly discovered for the bottleneck workplaces laser, bending and milling. The technology group turning is also highly occupied, but only with steady workload without any big resulting queues.
- Bad availability results for bending workplaces because of high setup rates.

As accomplished in reality, the resulting bottlenecks at laser and bending workplaces are eliminated by working extra shifts on weekends.

This chapter concludes the V&V task model implementation including a validated executable model as result. The gained knowledge about the system's behaviour in quarter 1 and 2 of 2022 will be compared in the following chapter 3.6 with a different product mix and production volume for quarter 1 and 2 of 2023.

3.6 Simulation Results

Having the simulation goal in mind by achieving insights about bottleneck properties in a job-shop, the previous chapter has identified characteristics and resulting bottlenecks of the year 2022. In this chapter the validated model will be used in order to analyse bottlenecks based on planned orders exported in December 2022 for the first half year of 2023 considering three different production volumes.

In general, it must be stated that while performing detailed scenario analysis and discovering some patterns in the data, it turns out that the exported data from the ERP system cannot be 100% compared to the input data in 2022. The findings are discussed with global process experts and lead to the following conclusions: First of all, the planned scenarios are a result of multiple MRP calculation runs of a global supply chain network including a **defined ruleset to determine the planned production order start dates**. That means for example, that the resulting release date for the production planner already includes a logic about how many orders should be released on different working days. In reality, the daily business of a logistics department implies to level those demands according to their available capacities. As first limitation, this levelling logic is not included in the simulation model, thus periodical peaks are expected to arise.

Secondly, the mentioned **MRP calculation** sometimes **summarises demands on a weekly basis**. Based on this rule, bigger lot sizes result than actually processed in reality, which has big influence on the sum of setup times that are stated in the scenarios. Additionally, when comparing the sum of setup times between 2022 and 2023, the deviation is even bigger than in reality because data inputs from 2022 include partly bookings, where multiple setup times are included.

The potential of not correctly considered setup times from 2022 because of partly bookings is prevented because of the following reason. Partly bookings into the ERP system are a common method in reality if operators can only confirm a certain portion of pieces belonging to one production order because of various reasons like shift change, missing material or machine breakdown. Every partly booking then includes a setup time based on the working plan, even if there is no changeover necessary. This potential deviation is avoided by the model, since production orders are created by production planner based on all related operations and setup is skipped at a workplace when the predecessor operation included the same material as the following one.

Nevertheless, the two speciality facts about the MRP calculation method represent the main reasons for a general drop of setup time in all scenarios. The impact of this change regarding setup time is, that mostly weekly lot sizes will be analysed in all 2023 scenarios. The rest of used target times is resulting from the actual demand of planned product mix and production volume between January and June 2023.

3.6.1 Scenario 1 representing 100%

The first analysed scenario uses production orders actually planned for quarter 1 and 2 in 2023, thus hereinafter referred to as 100% scenario. As shown in Table 4, this scenario includes for the same period of 6 months 22% less setup time, but about 5% more process time resulting from changed lot sizes and different product mix and production volume. The effect on unique technology groups or workplaces will be discussed in this chapter in detail below Figure 23, summarised in Table 5. The big drop regarding setup time was explained in chapter 3.6 and can be summarised as lot size logic change resulting from a global ERP system's MRP calculation.

Name	∆setupT [h]	∆setupT [h%]	∆processT [h]	∆processT [%]
100% scenario	-4200	-22	+4700	+5

Table 4: input values 100% scenario 2023 compared to 2022

As one output of the described VSD workshop in chapter 3.2, the target processing times for straightening have been reduced, thus straightening performance level is set to 100% in the workplace input file. As a test, milling time adaptations are taken out as well, which results in a constant WIP increase, similar but not as extreme as visualised in Figure 19. Still, especially for MF024 a queue results with unrealistic mean waiting times of 185 hours. Thus, a similar situation is present regarding data uncertainties described in the validation process including the three mentioned possibilities above Figure 19.

After adjusting bending machines regarding setup time, straightening process time and processing time for two milling workplaces, the WIP development looks as printed in Figure 23.

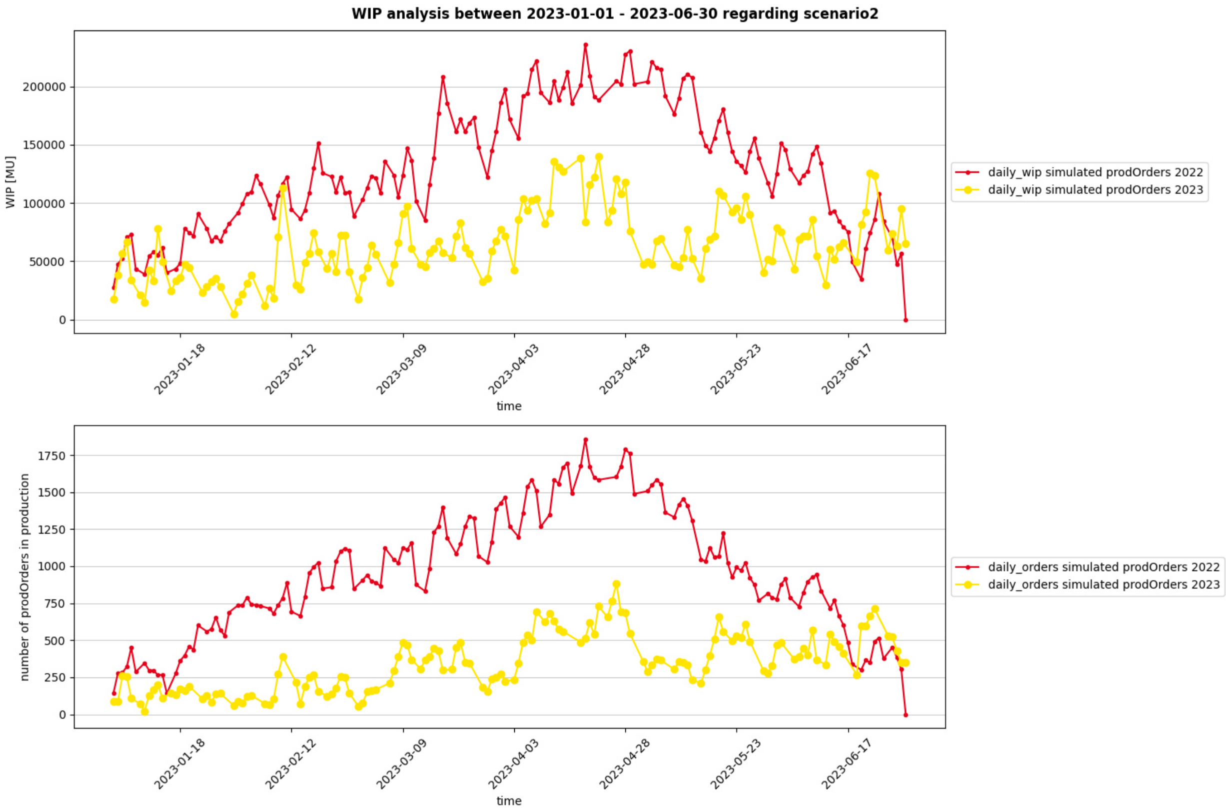

Figure 23: WIP development 100% scenario in 2023

Figure 23 shows the development of WIP for this scenario representing 100% planned production volume of 2023. Due to the difference product mix, different bottlenecks arise, summarised in Table 5.

bottleneck (WPL/techn.)	Δprocess time [%]	mean waiting time [h]	max. waiting time [h]	occupancy rate [%]
MF024	Same	43	164	76
MFC042	+10	37	149	84
Turning	+18%	35	157	91
MRB004	-23%	19	131	81

Table 5: bottleneck summary 100% scenario 2023

MF024 has about the same planned workload in 2023 like in 2022, because of that it still needs time adaptions meaning that either other workplaces will have to take over orders or overtimes additionally to 3 shifts will be necessary. Due to less setup time the simulated occupancy rate drops from 88% to 76%. As another milling machine, MFC042 was fine without adaptation factors in 2022, but in 2023 10% more processing hours await this workplace. Therefore, overtimes will be necessary, or possibilities established to enable different workplaces to help out.

Laser workplaces are not that big bottleneck anymore, average waiting times in this technology queue decreased from 34 hours to 26 hours. That may sound a bit high, but lasers are working in three shifts which means that material over the weekend will rise that averaged duration. The queue does not rise that much mainly because the additional new laser MYXL026 is working all time, thus not that many overtime shifts will be necessary compared to 2022. The occupancy rate stays high at about 85%.

As expected, straightening results -23% of process time because target times were reduced by the local team. Nevertheless, additional time adaptations of target times are necessary to avoid an accumulating queue, meaning that not every part will be able to be straightened.

In 2023 as problematic identified workplaces are related to the technology group turning. There, an increase of process times by 18% is found in this scenario. Those additional 1900 process hours will not be able to be produced with the same shift models as in 2022. Some turning workplaces can be extended to 4 shifts, if not all orders can be handled by those capacities, workload may be needed to be outsourced or started earlier to level high workload periods. Queue mean waiting times increased from 17 hours in 2022 to 35 hours in 2023. As mentioned at the end of chapter 3.5.2, turning workplaces were already highly occupied in 2022, those additional process increase the occupancy rate to 91% in mean, which leads to longer waiting times.

As an additional visual bottleneck analysis method, Figure 24 shows the waiting time range of all mentioned queues from above. By adding a simulation time attribute to production orders in the model, it is possible to track the exact waiting time of each order in every queue. The meaning of boxplots is explained in the end of chapter 2.4.3. It can be analysed that especially MF024, MFC042 and turning have big waiting times which cause high WIP and lead time. The queue waiting time is recorded every 2 hours, if any operator is assigned, resulting in over 1800 data points.

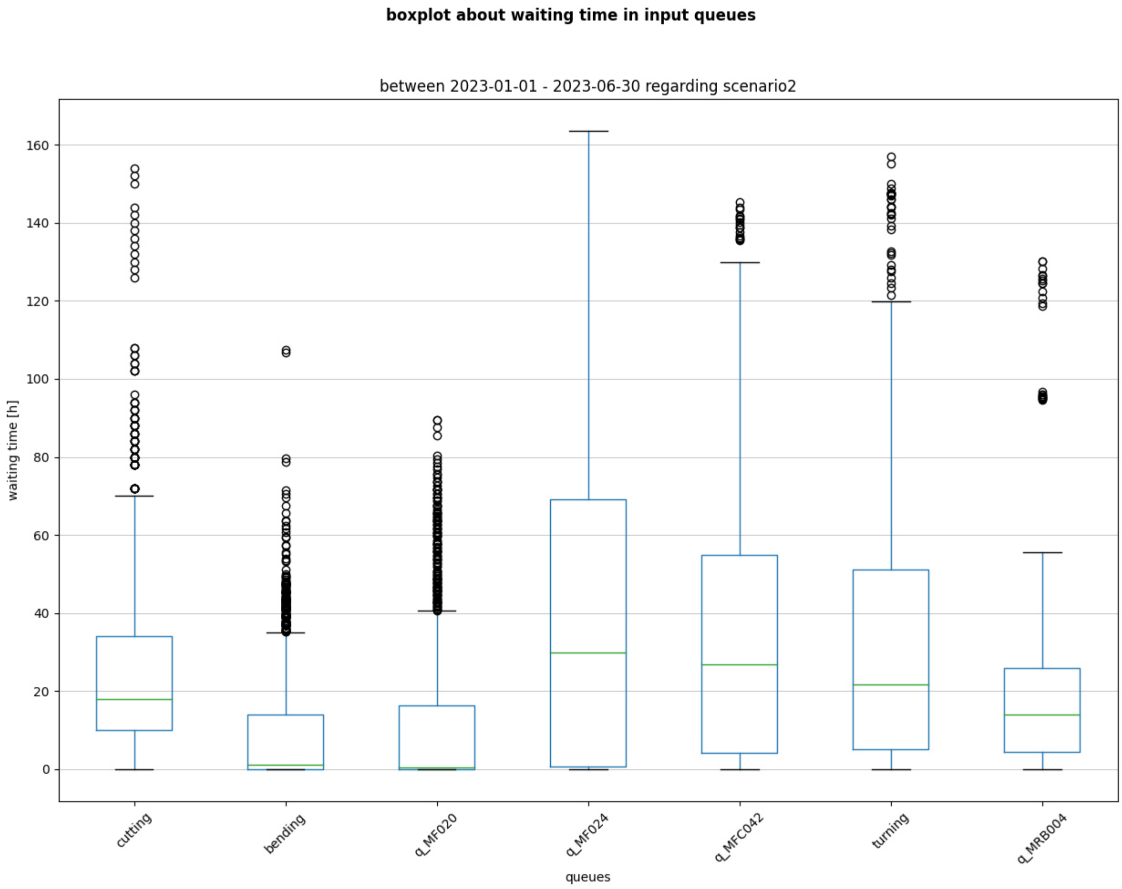

Figure 24: bottleneck queue waiting times scenario 100% 2023

Based on the planned scenario, bending and the milling workplace MF020 won't result as stationary bottlenecks. The scenario states 20% less setup for bending, but 16% more process time, which results in a far less queue length. That's because setup times are quite long for big bending machines and a decrease enables far more throughput. According to the scenario not that many overtime shifts will be necessary compared to 2022. By that, mean occupancy levels change from mean 85% to 73%. An additional fact resulting to that conclusion is that the new bending press MK037 from 2022 is fully productive in 2023. MF020 has less planned workload in 2023, thus no time adaptations are necessary, mean waiting time drops from 50 hours to 14 hours and the occupancy rate drops from 93% to 82%. Based on the different product mix and production volume the need of this workplace decreased by 56% in setup and 33% in process time compared to 2022. Nevertheless, recognisable in Figure 24 MF020 sometimes has long waiting times for selective production orders which is an indication for a dynamic bottleneck. The same evidence is identifiable for lasers, straightening and bending.

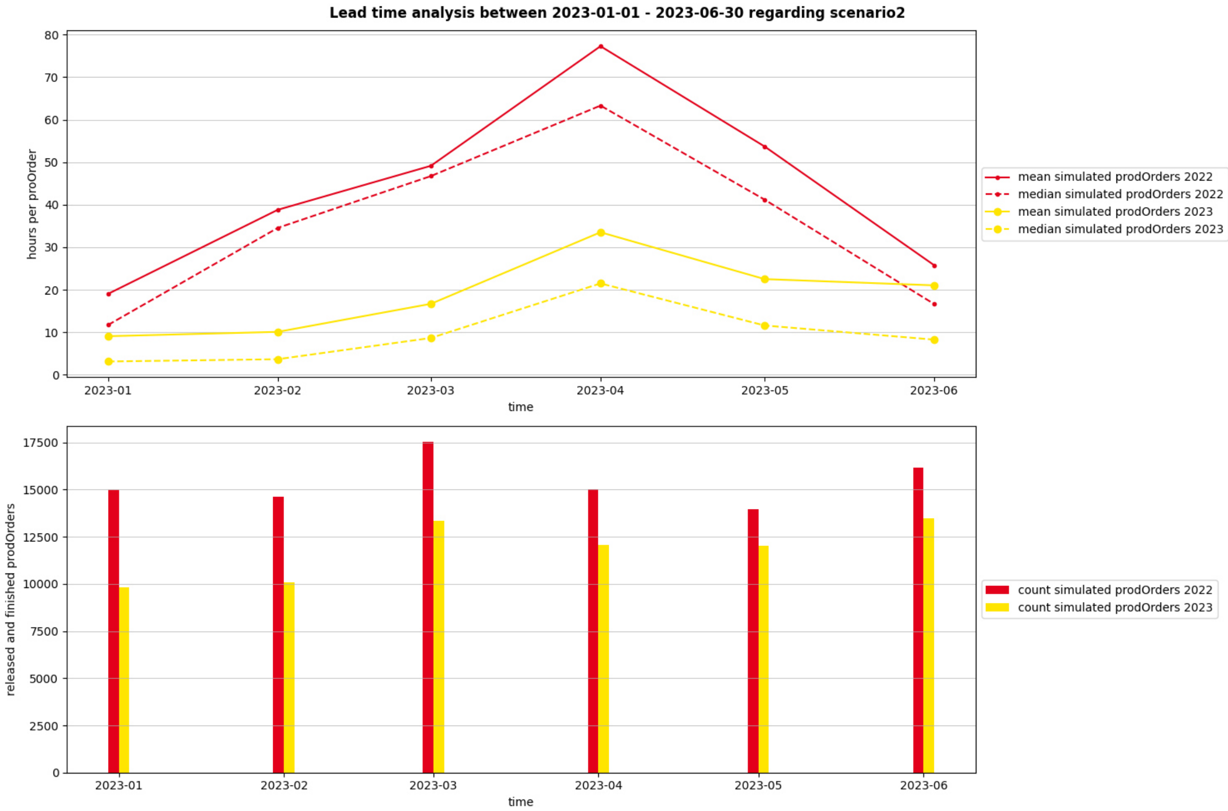

Figure 25: lead time development 100% scenario 2023

Recapitulated, due to new workplaces started in Q2 in 2022, the technology groups laser and bending won't be affected as bottlenecks that much as in 2022. That means that less overtimes will be necessary and faster lead times will be achieved. Due to the general change of bigger lot sizes represented in the scenarios, less WIP will result in the job-shop, mainly because of smaller lead times resulting from shorter queue-waiting times for most workplaces, shown in Figure 25. The mean maximum waiting time over all queues is 114 hours which might seem high, but from 121 hours of the validated model of 2022 can be concluded that the system is able to handle the workload including the mentioned proposals for critical workplaces. Important to state for this last conclusion is the fact, that the whole job-shop is inactive over Easter for 4 full days which represent 96 hours of additional waiting time in a queue as a consequence of public holidays.

3.6.2 Scenario 2 representing 110%

Originally it was planned or thought to be interesting by analysing a 125% scenario. But as the 100% scenario already includes 5% more process time compared to 2022, which has already been a production system with some resulting bottlenecks, it turns out that such a scenario overloads the system. Thus, especially lasers are highly overwhelmed that there are 23.000 production orders left in the input queue, even if 5 lasers do 46 overtime shifts each, that the rest of the job-shop can't be evaluated reasonably.

Thus a 10% surplus scenario is chosen, which results with 19% more process time and 19% less setup time compared to 2022, summarised in Table 6. This scenario includes 13% more process time and 3% more setup time compared to the 100% scenario. In order to compare both scenarios, no changes regarding target time adaptations are made.

Name	∆setupT [h]	∆setupT [h%]	∆processT [h]	∆processT [%]
110% scenario	-3800	-19	+15800	+19

Table 6: input values 110% scenario 2023 compared to 2022

In this scenario, lasers are very busy and 5 of them have to do 39 overtime shifts each. WIP of the job-shop develops like shown in Figure 26. It shows obviously how WIP constantly increases, affecting the WIP sum as also the sum of orders in the production system.

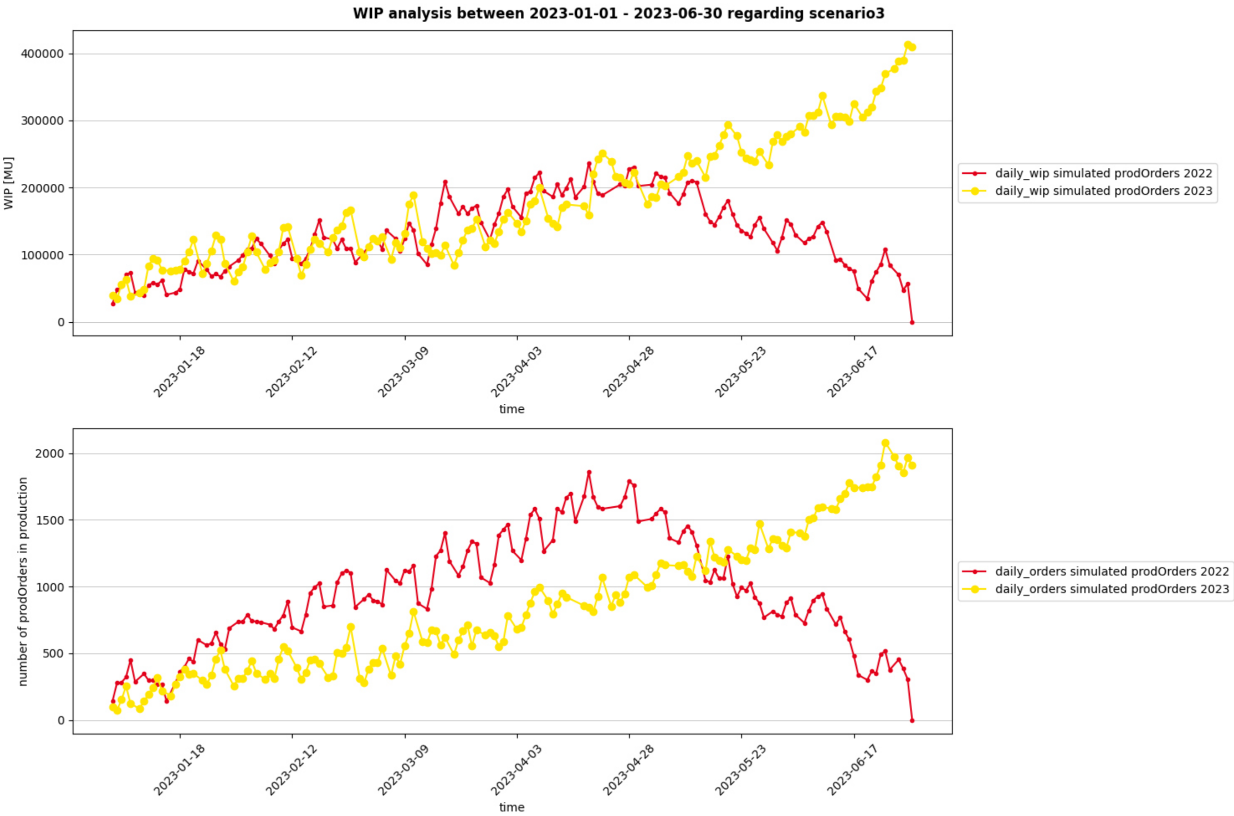

Figure 26: WIP development 110% scenario 2023

The main drivers for this constant WIP increase are summarised in Table 7.

bottleneck (WPL/techn.)	∆process time [%]	mean waiting time [h]	max. waiting time [h]	occupancy rate [%]
MFC042	+25	155	426	89
turning	+35	110	294	96
cutting	+23	102	269	89
MRB004	-10%	92	190	88
MF024	+6	46	164	79

Table 7: bottleneck summary 110% scenario 2023

As shown in Table 7, MFC042, which already results in the previous scenario as bottleneck, has a very long queue in this scenario and is the main reason for the constant WIP increase shown in Figure 26. If this simulated workload occurs in reality, other workplaces must assist or overtime shifts on the weekend have to be scheduled. Otherwise, the additional 980 hours of process time cannot be handled.

As second bottleneck technology turning is highly pressured with additional 3800 hours of process time, resulting in a long queue. As already identified in scenario 1, mean queue waiting times for a production order increase from 35 to 110 hours and mean amount of production orders in the queue rise from 46 to 159 orders. That amount will not be workable without additional resources like an additional workplace or outsourcing. The occupancy rate increased from 90% to 96% which means that the workplaces are really working on their limit, but still at the end of the simulation a big queue remains in front of the technology. Longest documented waiting time of an order exceeds 290 hours, which represents a critical bottleneck and results in very long lead times.

Third biggest queue are laser workplaces, those increased from 26 hours to 102 hours mean waiting time, although 5 lasers did 39 overtime shifts each. This high numbers lead to an increase of 23% of process time as well compared to 2022. Those additional processing hours cause a mean occupancy rate of 89% instead of 84. Including the additional new MYXL026, this amount of process times won’t be manageable without regular weekend shifts.

Fourth longest queue has straightening workplace MRB004 with 92 hours mean waiting time. Like already in the previous scenario, too many process hours are planned for this workplace, even if they result in 10% less compared to 2022. As this workplace is already working 4 shifts, outsourcing will be necessary if this huge amount of workload actually needs to be straightened. Occupancy rate is high with 88% and increase compared to the 100% scenario by 7%.

Fifth and last workplace with a critical queue waiting time over 24 hours is related to workplace MF024. Production orders have to wait 46 hours in mean to get processed. Occupancy rate stayed about the same as well, as processing time is only 1% higher compared to the scenario described in chapter 3.6.1. As MF020 can assist this workplace and is only planned for 2 shifts, the workload will be workable by doing some overtime and support the bottleneck workplace MF024.

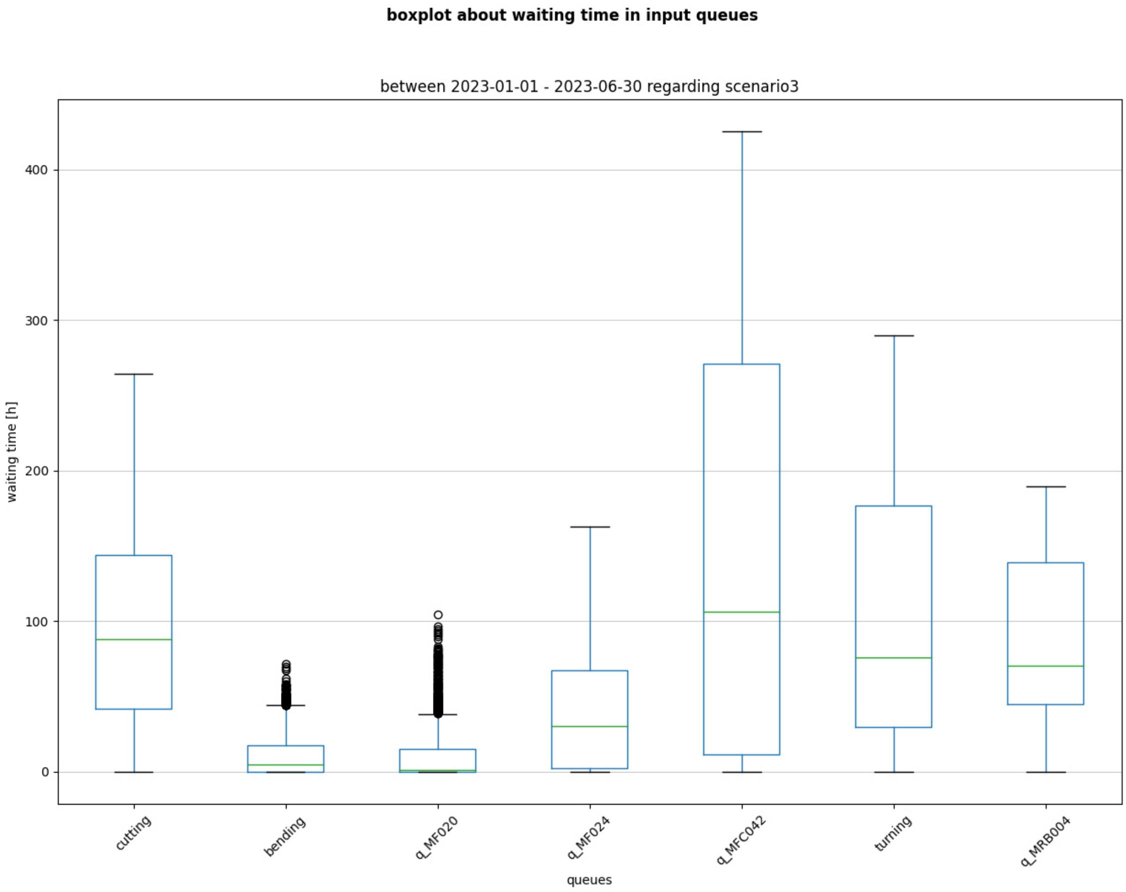

Figure 27: bottleneck queue waiting times scenario 110% 2023

Additional shifting bottlenecks can be identified in Figure 27, such as MF020 and bending infrequently have some outlier waiting times higher than 50 hours, but not comparable to the mentioned bottlenecks above. All other workplaces and technology groups are not identified as static bottlenecks of the whole period.

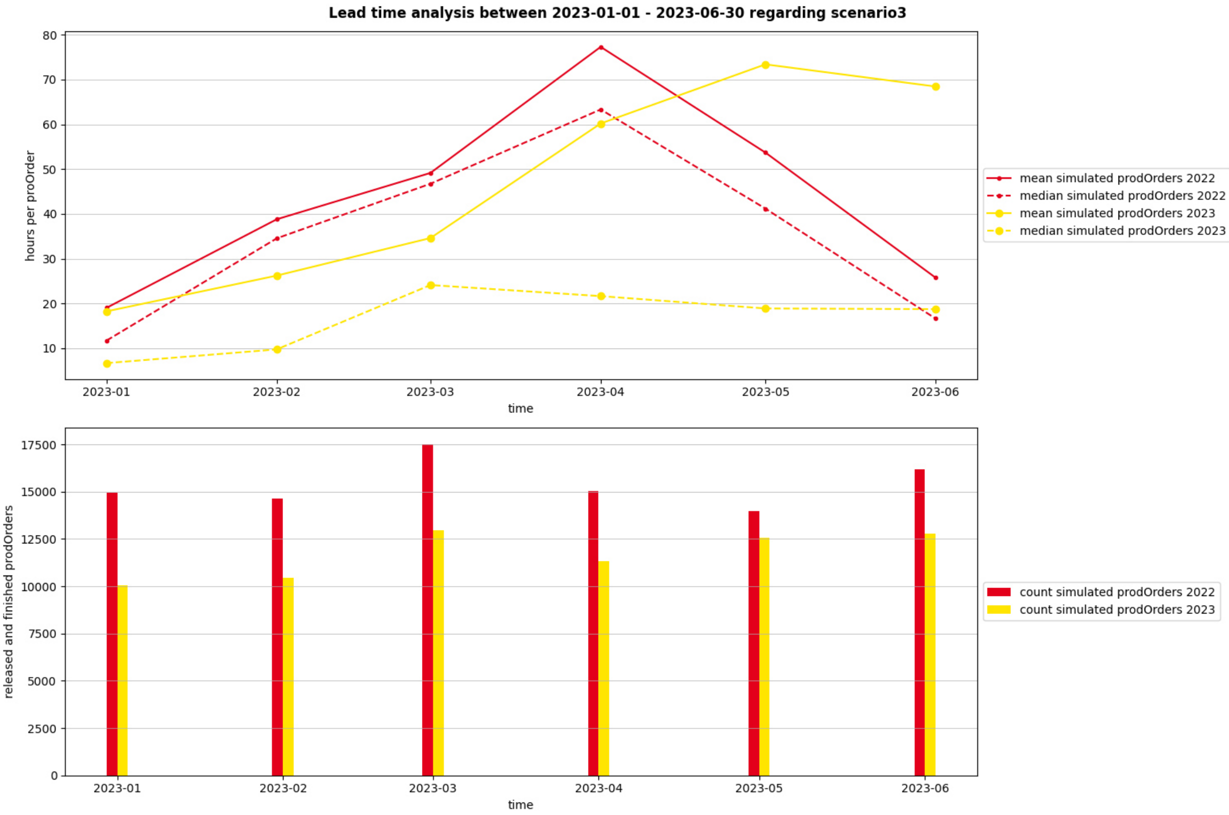

Figure 28: lead time development 110% scenario 2023

Focussing on the dotted median line in Figure 28 shows, that only a certain number of production orders is responsible for a high mean lead time. Knowing that especially the mentioned workplaces in Table 7 have high waiting times in their queues confirm the statement that especially those production orders rise the mean lead time.

The mean maximum waiting times over all queues results in 150 hours for this scenario. This value will lead to a by far more filled production system without the mentioned proposals, than the production system in 2022 having 121 hours overall average.

3.6.3 Scenario 3 representing 75%

The last discussed scenario handles a case that the production volume drops by 25%. The included processing time is 22% and setup time 34% lower compared to 2022, summarised in Table 8.

Name	∆setupT [h]	∆setupT [h%]	∆processT [h]	∆processT [%]
75% scenario	-6900	-34	-18500	-22

Table 8: input values 75% scenario 2023 compared to 2022

In comparison to the 100% scenario, process times are reduced by 26% and setup times by 16%. Thus, an underutilised production system results, shown in Figure 29. The workload of production orders is above average at the beginning and the end and can be recognised in peaks.

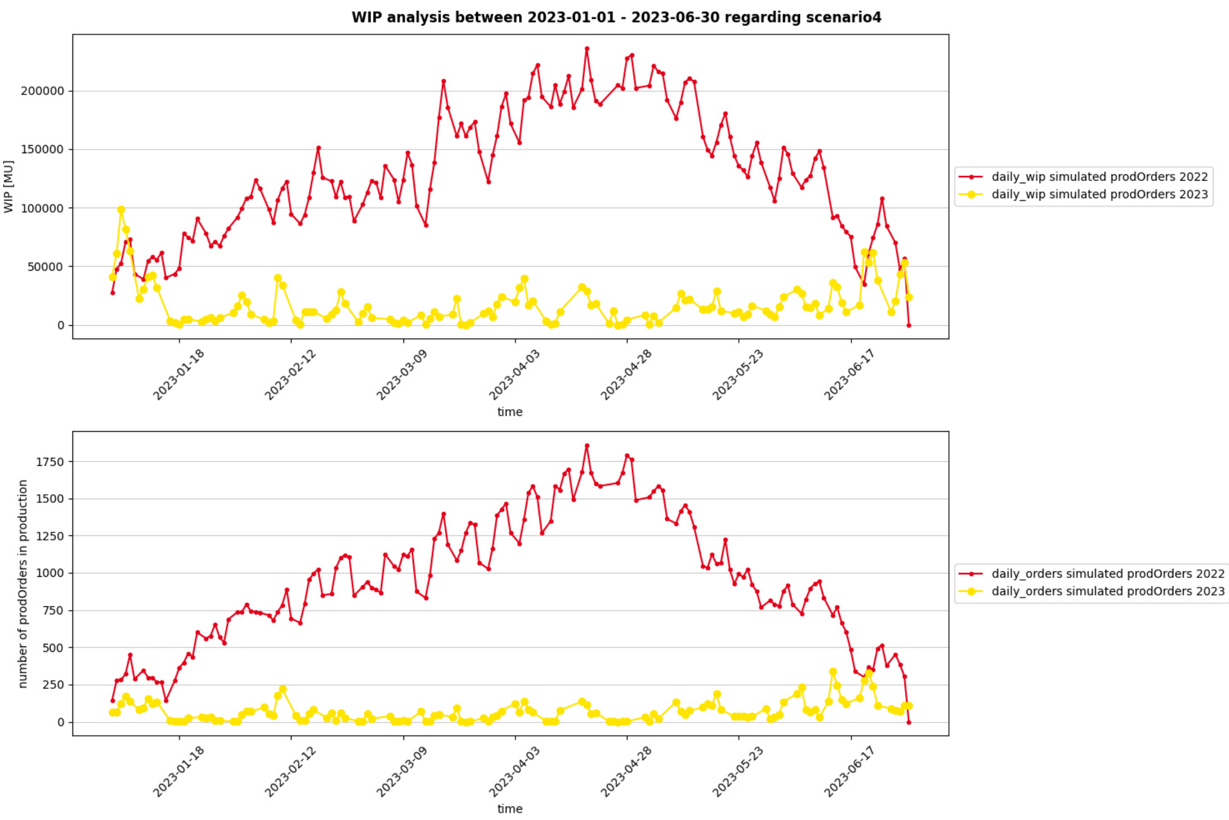

Figure 29: WIP development 75% scenario 2023

Evaluating this scenario with the same shift models like the previous mentioned scenarios, remarkable findings are the autogenous cutter with the longest mean queue, having 15 hours of mean waiting time. That is because this workplace has a high breakdown rate, consequently, production orders tend to stay longer in this queue. Turning machines and lasers turn out to have the highest mean occupancy rate of about 65%. Especially bending workplaces have a short mean waiting time because of low setup rate, occupancy rates are about 60%.

As expected, a scenario representing 25% less production volume the production system will be rather empty, lead times drop from a mean of 15 hours in the 100% scenario to five hours in the 75% scenario. In reality, such circumstances will lead to an adaptation of shifts to the actually needed capacities.

To evaluate the needed level of reduced capacities, according to the low resulting occupancy rates, the planned shifts of all workplaces are reduced. It turns out that in general nearly all planned shifts can be reduced by one total shift. A very low occupancy rate at milling and chamfering workplaces allow to go down from 4 to 2 shifts. 2 of 7 lasers have to continue working in 3 shifts, otherwise too long waiting queues result, and a lot of additional unplanned shifts would be necessary.

The outcome can be seen in Figure 30. The general reduction of shifts needs some overtime shifts at lasers, bending and turning. As already slightly noticeable in Figure 29, at the beginning and ending of the examined scenario, above average process times are planned for production orders.

Figure 30: WIP development 75% 2023, adapted shifts

The increase of WIP in the end is explainable by a higher workload for MFC042, turning, PSFB and the mentioned bottlenecks in Table 9, which are not able to keep up and developing an increasing queue in front of them. Those outliers can also be detected in Figure 31.

The dominating bottleneck workplaces after shift adaptions are summarised in Table 9Table 7.

bottleneck (WPL/techn.)	∆process time [%]	mean waiting time [h]	max. waiting time [h]	occupancy rate [%]
MF024	-38	42	287	75
MF020	-48	41	185	99
chamfering	-16	35	140	88

Table 9: bottleneck summary 75% scenario 2023

MF024 and MF020 are reduced by one shift, this reduction results in 4 overtime shifts for MF020, which has more production orders with high waiting time, shown in Figure 31. Additionally, a high occupancy rate shows the high workload for this workplace if it is productive for 1 shift less. MF024 faces an unusually large number of production orders at the end of the simulation period, which results in a long queue and high WIP. Chamfering workplaces are normally planned for 4 shifts, by reducing them to 2 shifts, a varying queue results that leads to a longer mean waiting time.

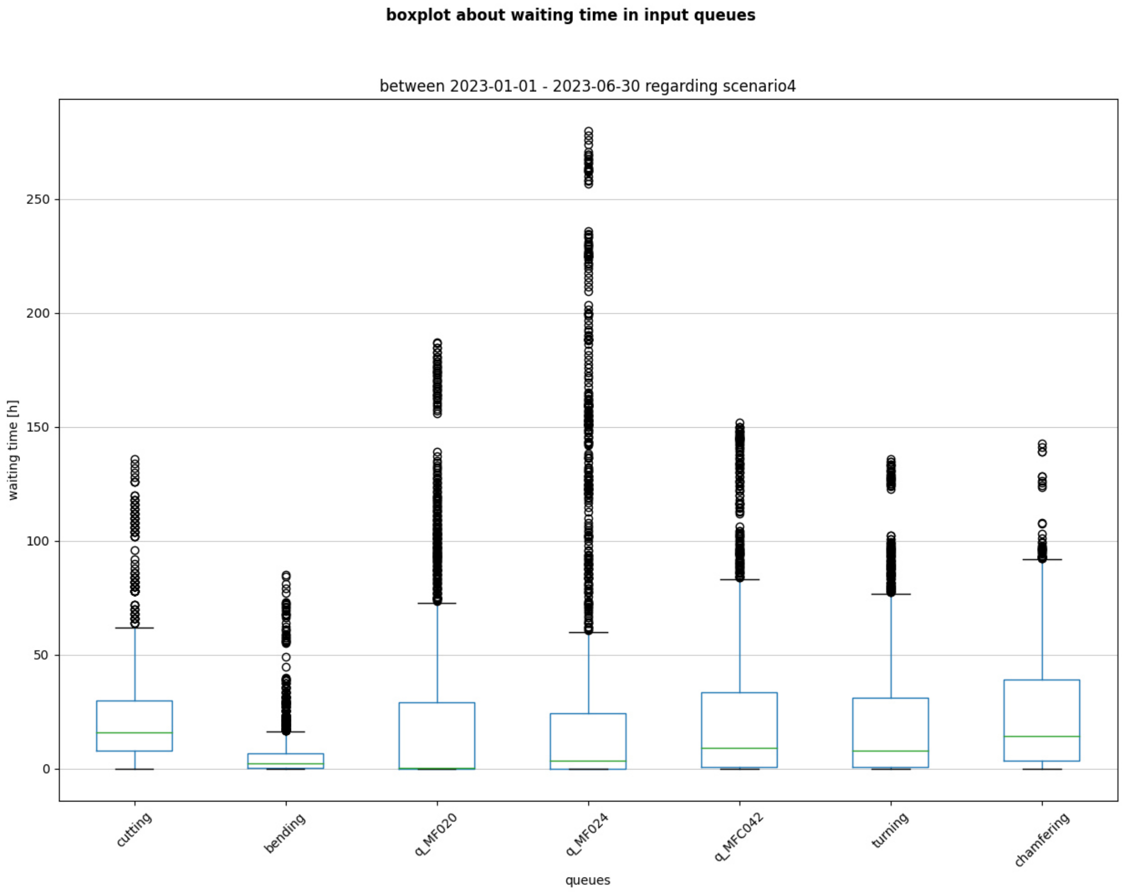

Figure 31: bottleneck queue waiting times scenario 75% 2023

This example scenario shows the general possibility of using a DES model to adapt shift times for production scenarios where shift amounts have to be changed and simulating the resulting effect.

Having the last scenario discussed, the empirical part of this thesis is finished. Except the determination of bottlenecks in a job-shop, additional examination use cases could be applied to such a DES model. Examples will be mentioned in the Outlook chapter 4.2.

4 Final Consideration

This final chapter summarises the present thesis and gives an outlook into further examinations possibilities.

4.1 Conclusion

This thesis investigated the possibility to identify bottlenecks including their characteristics in a job-shop production environment using discrete-event simulation (DES). Currently, common methods used to detect limited capacities include VSM or straightforward utilisation analysis approaches. Applying VSM is simple and fast but either evaluates an isolated current view or periodical mean process times, assuming that every working day is the same. Therefore, it is not possible to determine peak queue lengths or shifting bottlenecks resulting of a dynamic production system. Lot sizes and the actual production mix have big influence on changing bottlenecks in a production environment. From a DES model, it is possible to retrieve mean values as well, or inspect any remarkable system's state by evaluating every timestep. Especially stochastic events heavily influence limited capacities of a changing production system, which cannot be taken into account when a constant system behaviour is assumed.

To evaluate a system's behaviour correctly and derive useful conclusions, it is essential to apply appropriate key performance indicators. Thus, this thesis further focused on KPIs that evaluate the characteristics of bottlenecks, such as the occupancy rate of a workplace and waiting times in its queue. Additional KPIs were introduced that help to improve the flow rate of a production system, such as an availability metric that includes occupancy rate, process availability and setup reduction.

DES is already a frequently used simulation method for assembly areas, processing lines or in other more standardised industries. The application in a multi-technology job-shop including a broad variety of end products in size and weight contains due to the lower level of standardisation certain challenges. To develop a model in similar accuracy compared to a more standardised work environment, a unique set of rules has to be provided to achieve useful results.

In order to conclude valid simulation statements regarding bottlenecks, a generic framework was set up in Python, enabling to import various data source compositions. Those inputs can be adapted, transformed, filtered or converted by the framework to have correctly adjusted and needed data inputs available for the simulation model.

The discussed theoretical chapters in the first half of this thesis were applied in the second half, where an exemplary job-shop of an industry partner got transferred into a virtual model. A production environment of 43 workplaces was investigated, covering 9 different technologies from laser cutting, deburring, chamfering, sawing, turning, drilling, milling, straightening until bending.

Stochastic distributed breakdowns got determined based on the documented repair times of the past 18 months, moreover the difference to deterministic values was discussed. As validation period 68 days from March to May 2022 were chosen and best model parameters determined. Therefore, booked WIP and lead time were compared between the reality and modelled results by applying two-dimensional t-test, RMSE, MAE and R-squared metrics. To achieve a verified and validated executable model, the V&V procedure model of Rabe, Spieckermann and Wenzel was used as reference.

This validated model was then used in order to determine potential bottlenecks based on planned scenarios of the ERP system for the first two quarters of 2023. In total, three scenarios got simulated including various production volumes and a different product mix compared to 2022. Additionally, the number of orders has changed, mainly resulting by a non-identical lot size logic resulting from the MRP calculation.

Based on those inputs, the validated model enabled to identify new evolved static and dynamic bottlenecks by analysing the development of waiting times of each queue in detail, as also interpreting mean and maximum values. These measures, in combination with the occupancy rate, allowed to propose necessary adaptations regarding shifts and alternative workplaces. Additional machines that started operation in the second quarter of 2022 changed the bottleneck situation generally, thus not that many overtime hours will be necessary. However, possibilities are pointed out to handle bottlenecks that arise because of higher workload at specific workplaces. Moreover, a decreased production volume scenario was analysed where the effect of shift adaptions was examined.

This simulation study faced the importance of correct master data, exemplary mentioning defined shifts of workplaces, documentation regarding overtime shifts, adjustment of working plans or definition of alternative workplace matrix concerning overlapping material capabilities and validity of target setup and processing times. Summarising, correct working plans in the ERP system are one of the biggest pre-conditions to ensure fast success of DES. Otherwise, an extensive manual data collection and preparation phase is necessary, further assumptions regarding missing data have to be made or simplifications need to be applied.

Summing up, this thesis used DES to identify static and shifting bottlenecks by using a combination of detailed queue waiting time range, mean and maximum values including evaluating the occupancy rate. Those KPIs constitute bottleneck characteristics worth paying attention. The validated model showed the applicability of DES in a job-shop production environment and allowed the simulation of scenarios including the derivation of proposals. That involves capacity planning and the adaptation of shift models. Especially because of currently emerging technologies like the Internet of Things (IoT), mentioned in the following and last chapter, a simulation-based approach will even offer greater potentials regarding short-term decision support.

4.2 Outlook

Digitalisation initiatives and actions regarding IoT will continue to boost the scope of simulation models, based on the additional reliable data that can be integrated. Taking constantly measured inventory or queue levels as an example, which will allow constant input and feedback and avoid a warm-up phase by starting from initial reality amounts. Simulation runs can then be performed regarding the upcoming hour, shift, day or week including a wide range of use cases, like adapting production schedules based on current workplace availability and resulting shifting bottlenecks. Another benefit is using feedback values from reality and continuously improving the used model by tuning its parameters.

Additionally, the presented model can be extended by quality data in order to include KPIs like OEE. Other enhancement opportunities include the consideration of all internal transport modes to evaluate different layout options and transport equipment. Furthermore, the inclusion of additional reporting events enables more detailed feedback about the present shopfloor status such as current workplace occupancy rates and allow deeper analysis of critical workplace sequences. Gained benefits are more detailed information about the influence of production order mixes and the possibility to derive knowledge which specific mixture or volume is not appropriated for the production system. A general optimisation algorithm regarding optimal lot sizes, queue priority rules, production control systems and adaption of shift amounts referring to needed capacities promises further extension potential.

Closing this thesis in 2023, currently many initiatives strive for realising a fully developed 'Digital Twin' that is able to analyse complex interdependencies, provide decision support or even take power of control. But first, we have to ensure achieving visibility and transparency into sensible processes, in order to understand why something really happens and then decide if modelled support is wanted and needed. Prediction and modelled conclusions are only possible if an accurate basis is available.

5 List of references

Acharya, S. (2021): What are RMSE and MAE? - Towards Data Science. URL: https://towardsdatascience.com/what-are-rmse-and-mae-e405ce230383 (viewed: 10.01.2023).

Balci, O. (2003): Verification, validation, and certification of modeling and simulation applications. In: Winter Simulation Conference 2003, S. 150–158. DOI: 10.1109/WSC.2003.1261418.

Banks, J. (1998): Handbook of simulation: Principles, methodology, advances, applications, and practice, New York: Wiley. ISBN: 9780471134039.

Banks, J.; Carson, J. S.; Nelson, B. L.; Nicol, D. M. (2013): Discrete-Event System Simulation: Pearson New International Edition. 5th Edition, Harlow: Pearson Education Limited. ISBN: 1-292-02437-2.

Bauer, J. (2017): Produktionscontrolling und -management mit SAP® ERP: Effizientes Controlling, Logistik- und Kostenmanagement moderner Produktionssysteme. 5., revised and updated edition, Wiesbaden et al.: Springer Vieweg. ISBN: 978-3-658-18365-3.

Bauernhansl, T.; Hompel, M. ten; Vogel-Heuser, B. (2014): Industrie 4.0 in Produktion, Automatisierung und Logistik, Wiesbaden: Springer Fachmedien Wiesbaden. ISBN: 978-3-658-04681-1.

Borshchev, A.; Filippov, A. (2004): From System Dynamics and Discrete-event to Practical Agent Based Modeling: Reasons, Techniques, Tools. St. Petersburg Technical University. 22nd International Conference of the System Dynamics Society, Oxford, England: 25-29 July 2004.

Box, G. E. P.; Draper, N. R. (1987): Empirical model-building and response surfaces. [7. Dr.], New York: Wiley. ISBN: 9780471810339.

Bracht, U.; Geckler, D.; Wenzel, S. (2018): Digitale Fabrik: Methoden und Praxisbeispiele. 2., aktualisierte und erweiterte Auflage, Berlin et al.: Springer Vieweg. ISBN: 978-3-662-55782-2.

Brailsford, S. C. (2014): Discrete-event simulation and system dynamics for management decision making, Chichester, West Sussex: Wiley. ISBN: 978-1-118-34902-1.

Chair of Mathematics and Statistics (2015): Skriptum zur Vorlesung Statistik, 7., korrigierte Auflage, Montanuniversität Leoben.

Chen, X.; Voigt, T. (2020): Implementation of the Manufacturing Execution System in the food and beverage industry. In: Journal of Food Engineering, Vol. 278, S. 109932. DOI: 10.1016/j.jfoodeng.2020.109932.

Einzinger, P. (2014): A Comparative Analysis of System Dynamics and Agent-Based Modelling for Health Care Reimbursement Systems. Technische Universität Wien (Ed.), Institut für Analysis and Scientific Computing. DOI: 10.34726/hss.2014.21299.

Fabri, M.; Ramalhinho, H.; Oliver, M.; Muñoz, J. C. (2022): Internal logistics flow simulation: A case study in automotive industry. In: Journal of Simulation, Vol. 16, No. 2, S. 204–216. DOI: 10.1080/17477778.2020.1781554.

Friederich, J.; Francis, D. P.; Lazarova-Molnar, S.; Mohamed, N. (2022): A framework for data-driven digital twins of smart manufacturing systems. In: Computers in Industry, Vol. 136, S. 103586. DOI: 10.1016/j.compind.2021.103586.

Galarnyk, M. (2022): Understanding Boxplots. URL: https://builtin.com/data-science/boxplot (viewed: 12.02.2023).

Georgiev, G. (2019): “All Models Are Wrong” Does Not Mean What You Think It Means. URL: https://medium.com/swlh/all-models-are-wrong-does-not-mean-what-you-think-it-means-610390c40c9c (viewed: 28.01.2023).

Ghaleb, M.; Zolfagharinia, H.; Taghipour, S. (2020): Real-time production scheduling in the Industry-4.0 context: Addressing uncertainties in job arrivals and machine breakdowns. In: Computers & Operations Research, Vol. 123, S. 1–44. DOI: 10.1016/j.cor.2020.105031.

Goldratt, E. M.; Cox, J. (1993): The Goal: A process of ongoing improvement. 2nd ed., Aldershot: Gower. ISBN: 0-566-07417-6.

Gómez Paredes, F. J.; Godinho Filho, M.; Thürer, M.; Fernandes, N. O.; Jabbour, C. J. C. (2022): Factors for choosing production control systems in make-to-order shops: a systematic literature review. In: Journal of Intelligent Manufacturing, Vol. 33, No. 3, S. 639–674. DOI: 10.1007/s10845-020-01673-z.

Gudehus, T. (2011): Dynamische Disposition, Berlin et al.: Springer. ISBN: 978-3-642-22982-4.

Günther, H.-O.; Tempelmeier, H. (2012): Produktion und Logistik. 9., updated and extended Ed., Berlin: Springer. ISBN: 978-3-642-25164-1.

Guo, H.; Chen, M.; Mohamed, K.; Qu, T.; Wang, S.; Li, J. (2021): A digital twin-based flexible cellular manufacturing for optimization of air conditioner line. In: Journal of Manufacturing Systems, Vol. 58, No. Part B, S. 65–78. DOI: 10.1016/j.jmsy.2020.07.012.

Gustavo, B.; Nacef, T.; Eric, C. (2019): Determining Production Systems Performance Metrics Considering Machine Downtime. In: IFAC-PapersOnLine, Vol. 52, No. 13, S. 1022–1027. DOI: 10.1016/j.ifacol.2019.11.329.

Gutenberg, E. (1969): Grundlagen der Betriebswirtschaftslehre: Die Produktion. 16. Edition, Berlin, Heidelberg et al.: Springer Berlin Heidelberg. ISBN: 978-3-662-37174-9.

Gutenschwager, K.; Rabe, M.; Spieckermann, S.; Wenzel, S. (2017): Simulation in Produktion und Logistik: Grundlagen und Anwendungen, Berlin et al.: Springer Vieweg. ISBN: 978-3-662-55744-0.

Hedtstück, U. (2013): Simulation diskreter Prozesse: Methoden und Anwendungen, Berlin et al.: Springer Vieweg. ISBN: 978-3-642-34870-9.

Huynh, B. H.; Akhtar, H.; Li, W. (2020): Discrete-event Simulation for Manufacturing Performance Management and Optimization: A Case Study for Model Factory. In: 9th International Conference 2020, S. 16–20. DOI: 10.1109/ICITM48982.2020.9080394.

Inamdar, A. (2020): Understanding Apache Parquet. URL: https://towardsdatascience.com/understanding-apache-parquet-7197ba6462a9 (viewed: 23.01.2023).

industryPartnerExperts (2022): Basic Layout.

Joppen, R.; Enzberg, S. von; Gundlach, J.; Kühn, A.; Dumitrescu, R. (2019): Key performance indicators in the production of the future. In: Procedia CIRP, Vol. 81, No. 52nd CIRP Conference on Manufacturing Systems, S. 759–764. DOI: 10.1016/j.procir.2019.03.190.

Kaiblinger, A.; Woschank, M. (2022): State of the Art and Future Directions of Digital Twins for Production Logistics: A Systematic Literature Review. In: Applied Sciences, Vol. 12, No. 669, S. 1–21. DOI: 10.3390/app12020669.

Klenner, F.; Lenze, D.; Schwarzer, S.; Deuse, J.; Friedrich, T. (2016): Smart Data Analytics zur Identifikation dynamischer Engpässe in Flexiblen Fertigungssystemen. In: at - Automatisierungstechnik, Vol. 64, No. 7, S. 540–554. DOI: 10.1515/auto-2016-0014.

Kritzinger, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. (2018): Digital Twin in manufacturing: A categorical literature review and classification. In: IFAC-PapersOnLine, Vol. 51, No. 11, S. 1016–1022. DOI: 10.1016/j.ifacol.2018.08.474.

Kulezak, M. (2022): The top 10 industrial software companies. URL: https://iot-analytics.com/industrial-software-companies/ (viewed: 31.05.2022).

Lang, S.; Reggelin, T.; Müller, M.; Nahhas, A. (2021): Open-source discrete-event simulation software for applications in production and logistics: An alternative to commercial tools? In: Procedia Computer Science, Vol. 180, S. 978–987. DOI: 10.1016/j.procs.2021.01.349.

Law, A. M. (2013): Simulation modeling and analysis. Fifth edition, Dubuque: McGraw-Hill Education. ISBN: 978-0-07-340132-4.

Lee, J.; Azamfar, M.; Singh, J.; Siahpour, S. (2020): Integration of digital twin and deep learning in cyber-physical systems: towards smart manufacturing. In: IET Collaborative Intelligent Manufacturing, Vol. 2, No. 1, S. 34–36. DOI: 10.1049/iet-cim.2020.0009.

Lindegren, M. L.; Lunau, M. R.; Mafia, M. M. P.; Da Ribeiro Silva, E. (2022): Combining Simulation and Data Analytics for OEE Improvement. In: International Journal of Simulation Modelling, Vol. 21, No. 1, S. 29–40. DOI: 10.2507/IJSIMM21-1-584.

Lödding, H. (2016): Verfahren der Fertigungssteuerung: Grundlagen, Beschreibung, Konfiguration. 3. Edition, Berlin et al.: Springer Vieweg. ISBN: 978-3-662-48458-6.

Matt, D. et al. (2021): Implementing Industry 4.0 in SMEs: Concepts, examples and applications, Cham, Switzerland: Palgrave Macmillan. ISBN: 978-3-030-70515-2.

Mattern, F.; Mehl, H. (1989): Diskrete Simulation - Prinzipien und Probleme der Effizienzsteigerung durch Parallelisierung. In: Informatik Spektrum, Vol. 12, No. 4, S. 198–210.

Molin, S. (2019): Hands-On Data Analysis with Pandas: Efficiently Perform Data Collection, Wrangling, Analysis, and Visualization Using Python, Birmingham: Packt Publishing Limited. ISBN: 978-1-78961-532-6.

Montgomery, D. C. (2009): Introduction to statistical quality control. 6th ed., Hoboken, N.J: Wiley. ISBN: 978-0-470-16992-6.

Morgan, J. S.; Howick, S.; Belton, V. (2017): A toolkit of designs for mixing Discrete-event Simulation and System Dynamics. In: European Journal of Operational Research, Vol. 257, No. 3, S. 907–918. DOI: 10.1016/j.ejor.2016.08.016.

Murphy, A.; Taylor, C.; Acheson, C.; Butterfield, J.; Jin, Y.; Higgins, P.; Collins, R.; Higgins, C. (2020): Representing financial data streams in digital simulations to support data flow design for a future Digital Twin. In: Robotics and Computer-Integrated Manufacturing, Vol. 61, S. 1–16. DOI: 10.1016/j.rcim.2019.101853.

Nyhuis, P. (2017): Hanoverian Supply Chain Model. URL: https://www.halimo.education/ (viewed: 13.02.2023).

Nyhuis, P.; Wiendahl, H.-P. (2012): Logistische Kennlinien: Grundlagen, Werkzeuge und Anwendungen. 3rd Edition, Berlin et al.: Springer Vieweg. ISBN: 978-3-540-92838-6.

Ōno, T. (2013): Das Toyota-Produktionssystem: Das Standardwerk zur Lean Production. 3rd extended and updated edition, Frankfurt et al.: Campus Verlag. ISBN: 978-3-593-39929-4.

Ostheimer, B.; Schickert, A. C. (2013): Mean Time Between Failures: Grundlagen, Konzept, Methoden, Werkzeuge. Professur BWL - Wirtschaftsinformatik, Justus-Liebig-Universität Giessen (Ed.), In: Arbeitspapiere WI, No. 4/2013. ISSN: 1613-6667.

Page, B.; Kreutzer, W. (2005): The Java simulation handbook: Simulating discrete-event systems with UML and Java, Aachen: Shaker. ISBN: 3-8322-3771-2.

Pawlewski, P.; Kosacka-Olejnik, M.; Werner-Lewandowska, K. (2021): Digital Twin Lean Intralogistics: Research Implications. In: Applied Sciences, Vol. 11, 1495, No. 4, S. 1–13. DOI: 10.3390/app11041495.

Pfohl, H.-C. (2018): Logistiksysteme: Betriebswirtschaftliche Grundlagen. 9th Ed. 2018, Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-662-56227-7.

Rabe, M.; Spieckermann, S.; Wenzel, S. (2008a): A new procedure model for verification and validation in production and logistics simulation. In: : Winter Simulation Conference 2008: IEEE. ISBN 978-1-4244-2707-9, S. 1717–1726.

Rabe, M.; Spieckermann, S.; Wenzel, S. (2008b): Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodelle und Techniken. 1st Edition, Berlin et al.: Springer. ISBN: 978-3-540-35281-5.

Reinhart, G. (2017): Handbuch Industrie 4.0: Geschäftsmodelle, Prozesse, Technik, München: Hanser. ISBN: 978-3-446-44642-7.

Roser, C.; Lorentzen, K.; Deuse, J. (2015): Reliable shop floor bottleneck detection for flow lines through process and inventory observations: the bottleneck walk. In: Logistics Research, Vol. 8, No. 1. DOI: 10.1007/s12159-015-0127-2.

Roser, C.; Shook, J. (2021): All about pull production: Designing, implementing, and maintaining Kanban, CONWIP, and other pull systems in Lean Production, Offenbach: AllAboutLean.com Publishing. ISBN: 978-3-96382-028-1.

Ruud van der Ham (2020): salabim's Google Group. URL: https://groups.google.com/g/salabim/c/lsPGYO6rIQk/m/wAXEz9cjAgAJ (viewed: 10.02.2023).

Sargent, R. (2010): Verification and validation of simulation models. In: Winter Simulation Conference 2010, 166-183. DOI: 10.1109/WSC.2010.5679166.

Schuh, G.; Anderl, R.; Dumitrescu, R.; Hompel, M. ten; Krüger, A. (2020): Industrie 4.0 Maturity Index. Managing the Digital Transformation of Companies. acatech - National Academy of Science and Engineering (Ed.). 2nd Edition.

SciPy community (2023a): scipy.stats.levene — SciPy v1.10.0 Manual. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.levene.html (viewed: 12.01.2023).

SciPy community (2023b): scipy.stats.ttest_ind — SciPy v1.10.0 Manual. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html (viewed: 12.01.2023).

Shou, W.; Wang, J.; Wu, P. (2021): The application of simulation in lean production research: a critical review and future directions. In: Engineering, Construction and Architectural Management, Vol. 28, No. 8, S. 2119–2154. DOI: 10.1108/ECAM-10-2020-0853.

Tao, F.; Qi, Q.; Wang, L.; Nee, A. (2019): Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. In: Engineering, Vol. 5, No. 4, S. 653–661. DOI: 10.1016/j.eng.2019.01.014.

Tempelmeier, H. (2018): Modellierung logistischer Systeme, Berlin et al.: Springer Vieweg. ISBN: 978-3-662-57770-7.

van der Ham, R. (2018): salabim: discrete-event simulation and animation in Python. In: Journal of Open Source Software, Vol. 3, No. 27, S. 767. DOI: 10.21105/joss.00767.

van der Ham, R. (2021): State — salabim 21.1.7 documentation. URL: https://www.salabim.org/manual/State.html (viewed: 20.01.2023).

VDI 3423:2011-08: Technical availability of machines and production lines.

VDMA 66412-1:2009-10: Manufacturing Execution Systems (MES) Kennzahlen.

Wedel, M.; Hacht, M. von; Hieber, R.; Metternich, J.; Abele, E. (2015): Real-time Bottleneck Detection and Prediction to Prioritize Fault Repair in Interlinked Production Lines. In: Procedia CIRP, Vol. 37, S. 140–145. DOI: 10.1016/j.procir.2015.08.071.

Wiendahl, H.-P. (2010): Betriebsorganisation für Ingenieure. 7th updated Ed., München: Hanser. ISBN: 978-3-446-41878-3.

Zhang, H.; Qi, Q.; Tao, F. (2022): A multi-scale modeling method for digital twin shop-floor. In: Journal of Manufacturing Systems, Vol. 62, S. 417–428. DOI: 10.1016/j.jmsy.2021.12.011.

Zsifkovits, H. E. (2012): Logistik. 1st Ed., Stuttgart et al.: UTB GmbH; UVK Lucius. ISBN: 978-3-8252-3673-1.