Automatic Verification and Validation of Automatically Generated Simulation-Based Digital Twins for Discrete Material Flow Systems

---

Author: Daniel Fischer

Supervisor: Prof. Christian Schwede

Program: Research Master Data Science

Institution: Hochschule Bielefeld (HSBI)

Submission Date: March 26, 2025

Abstract

This thesis investigates the application of Digital Twins within discrete material flow systems, a key component of Industry 4.0. It reviews the progression from Digital Models and Digital Shadows to fully realized Digital Twins that integrate real-time data, simulation, and control mechanisms. To learn these Digital Twins from data enables companies to reduce twin creation and updating efforts. These gained advantages would be made obsolete if the validation and verification of such twins were performed manually involving experts. To address this problem, this research proposes a data-driven framework for automated verification, validation, and uncertainty quantification of simulation-based DTs. Leveraging object-centric event logs and advanced machine learning techniques, the framework aims to seamlessly integrate digital simulation with real-world data. This approach ensures that the resulting DTs maintain high fidelity and robustness by continuously benchmarking their performance against operational metrics. The framework is evaluated via a comprehensive case study examining its effectiveness in enhancing process efficiency and predictive precision.

Keywords: Digital Twins, Automated VVUQ, Process Mining, Machine Learning, Discrete Material Flow Systems.

Contents

List of Abbreviations

AGDT Automatically Generated Digital Twin. 1

CPS Cyber-Physical System. 1

DES Discrete-Event Simulation. 1

DM Digital Model. 1

DMFS Discrete Material Flow Systems. 1

DS Digital Shadow. 1

DT Digital Twin. 1

IoT Internet of Things. 1

ML Machine Learning. 1

PM Process Mining. 1

PPC Production Planning and Control. 1

SBDT Simulation-Based Digital Twin. 1

V&V Verification and Validation. 1

VVUQ Verification, Validation, and Uncertainty Quantification. 1

Chapter 1

Introduction

1.1 Initial Situation

Digital Twins (DT) are a key technology at the front of the fourth industrial revolution, often referred to as Industry 4.0. The latter term is characterized by the integration of cyber-physical systems (CPS), the Internet of Things (IoT), and cloud computing to create smart factories aimed at automation and efficiency (Oztemel & Gursev, 2020). Companies pursue this vision by trying to remain competitive through the adoption of innovative technologies that promise enhanced productivity and reduced operational costs. One such technology that supports this transformation is the DT. It can be defined as a virtual representation of physical assets enabling real-time monitoring and optimization (Tao et al., 2018). The DT bridges the connection between the two entities with a bi-directional data flow to exchange information and to influence the behavior of the physical asset (Grieves, 2014). This technology is central to Industry 4.0, facilitating the physical and digital worlds through real-time data integration, simulation, and optimization (Judijanto et al., 2024).

Although this discipline is rapidly evolving, a unified definition of DT has yet to be established due to the diverse requirements and perspectives across different fields. In engineering, the focus might be on the real-time interaction between physical systems and their digital counterparts, whereas in computer science, the emphasis is often on data integration and simulation capabilities. These varying priorities result in multiple interpretations and applications of the term DT. The concept was first introduced by Michael Grieves in 2002, who defined it as a digital representation of a physical object or system (Grieves, 2014). However, the concept has evolved since, encompassing a broader range of applications and technologies. In the literature, three termes are used to describe similar characteristics of DT: Digital Model (DM), Digital Shadow (DS), and Digital Twin (DT), see Figure 1.1 (Jones et al., 2020; Zhang et al., 2021).

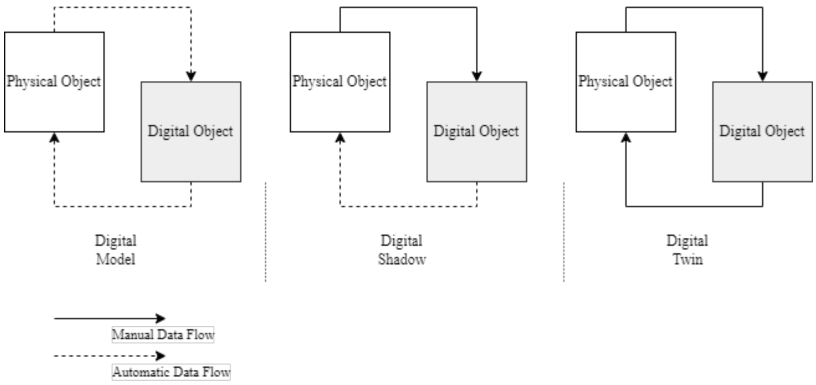

Figure 1.1: Comparison of Digital Shadow (DS), Digital Model (DM) and Digital Twin (DT) as presented by Kritzinger (2018). This distinction is crucial for understanding validation requirements across different digital representation types.

The Digital Model (DM) represents the most basic form. It involves manual data connections between physical and digital entities. These connections can be temporarily shifted or even disconnected. There is no direct control of the digital object over the physical entity. It is primarily a simple or complex model *describing* the physical object. The data flow must be manually triggered by the modeler, who also interprets the results and controls the DM. The Digital Shadow (DS) is a more advanced version of the DM. It is a digital representation of the physical object that is continuously updated with real-time data, allowing for monitoring, analysis and simulation. While it can predict future states of the physical object based on the current state and historical data, it is not able to influence the physical object without human intervention. The control is, similar to the DM, still in the hands of the modeller. A DS is frequently used for simulation purposes and is sometimes misclassified as a DT in the literature (Kritzinger et al., 2018; Sepasgozar, 2021). The Digital Twin (DT) is the most advanced version of the three, offering a digital representation of the physical object, which is also continuously updated with real-time data. The DT can be used for monitoring, analysis, and *control* purposes. It can predict the future states of the physical object based on the current state and historical data. The DT can also influence the physical object by sending control signals to it. The control is partially or completely in the hands of the DT. The DT thus *can* serve more purpose than modelling or simulating the physical object. It may serve as an autonomous system, updating itself or with minimal human intervention (Kritzinger et al., 2018).

DTs are applied across various sectors, including manufacturing, defense, automotive, service, finance and healthcare (Tao et al., 2018). Manufacturing is particularly notabl due to its high potential for process optimization and automation. This thesis focuses on the latter, particularly discrete material flow systems (DMFS). These systems process discrete objects (parts) moving along transportation routes or conveyor lines at regular or irregular intervals, integrating both production and logistics operations (Arnold & Furmans, 2005; Schwede & Fischer, 2024). A key simplification in their modeling is the abstraction of material flow as a sequence of discrete events, following the principles of discrete-event simulation (DES) (Kovacs & Kostal, 2016; Robinson, 2014). DES is well-suited for analyzing complex systems where state changes occur at discrete points in time, such as arrivals, departures, and processing steps (Robinson, 2014).

Historically, DM played a crucial role in the design, planning, and control of DMFS, primarily through applications like material flow simulations, logistic assistance systems, and digital factory implementations (Thiede et al., 2013). However, advancements in both DS and DT have enabled a shift from isolated, use-case-specific models toward complete digital representations that span the entire lifecycle of DMFS (Abdoune et al., 2023). This transition is largely driven by the growing demand for predictive capabilities by stakeholders and automated decision support in manufacturing systems, reflecting the core principles of Industry 4.0 (Frank et al., 2019). A second driver of DT innovation lies in the widely available data from IoT devices and sensors, which enhances model training and real-time adaptation of DTs (Tao et al., 2018).

In practice, the automated data transfer between the digital model and the physical system is not always critical for DMFS management. Unlike in time-sensitive applications, human decision-makers often remain integral to the control loop, meaning that real-time automation is not always necessary (Schwede & Fischer, 2024). Therefore, for this thesis, DS and DTs will be treated as equivalent concepts.

Beyond replicating the current state and managing historical data, DTs are essential for predicting system behavior and evaluating potential modifications. The widespread use of DES within digital twins highlights the central role of simulation-based DTs (SBDTs) in DMFS (Lugaresi & Matta, 2021). As Schwede and Fischer emphasize, SBDTs provide decision support for optimizing costs and performance in highly competitive manufacturing environments. While current SBDTs are primarily developed and updated manually by domain experts, emerging research explores how machine learning (ML) can enhance predictive accuracy and automate model updates by automatically learning model characteristics, reducing costs and development time.

Thus, the progression from digital models to simulation-based DTs reflects an ongoing shift toward data-driven, predictive, and increasingly automated representations of DMFS, enabling more informed decision-making throughout the system's lifecycle (Boschert & Rosen, 2016; Lim et al., 2020).

1.2 Problem

Despite the transformative potential of DTs, their implementation can be challenging. Creating and maintaining accurate DTs require substantial investments in technology and domain knowledge. This investment is wasted if the resulting model fails to accurately represent the physical entity or produces incorrect results. While automatic generation may seem like an elegant solution, it carries risks such as overfitting or biased predictions (Geman et al., 1992). Manufacturing data for training must be rigorously cleaned and preprocessed. Automatically generated DTs must also undergo automatic Validation, Verification, and Uncertainty Quantification (VVUQ) to preserve their cost and time advantages. Manual VVUQ, which relies on humans in the loop, hinders scalability, automatic synchronization with the physical entity, and depends on costly domain knowledge often provided by experts (Bitencourt et al., 2023). These hurdles are significant barriers to automatic learning (Ribeiro et al., 2016; Zhao et al., 2024). As industries integrate DT into their production processes, establishing trust becomes fundamental as well (Arrieta et al., 2020; Trauer et al., 2022). For widespread acceptance among coworkers, stakeholders, and investors, automatic DT creation and VVUQ must demonstrate clear advantages over manual creation and expert-led VVUQ.

Even when DT learning is successfully performed, questions about its correctness, precision, and robustness persist. These concerns are addressed by validation, verification, and uncertainty quantification frameworks (VVUQ) (Sel et al., 2025). Ensuring the validity, reliability, and accuracy of a DT is critical, yet traditional VVUQ approaches rely heavily on manual expert involvement and case-specific reference values (Bitencourt et al., 2023; Hua et al., 2022). This leads to inefficiencies, particularly in the context of automated DT generation, where such manual processes undermine the goal of reducing development effort. Hua et al. even argue that there are no robust and standardized verification and validation methods for DTs. As (Sel et al., 2025) point out, uncertainty quantification is often overlooked, but addresses an important aspect of assessing low noise in explanations. One hurdle to standardized VVUQ frameworks is the lack of a clear definitions for validity and verification in the context of DTs (Bitencourt et al., 2023).

For discrete material flow systems (DMFS), these challenges are even more pressing due to their procedural nature and inherent stochasticity. Rigorous VVUQ is essential to address the risk of manufacturing process failures caused by anomalies, resource constraints, software faults, or human error. This necessity arises because such failures can disrupt the intricate workflows and unpredictable dynamics inherent in DMFS, making reliable performance prediction a priority. When DTs for these systems are generated automatically, traditional validation methods become problematic, as they negate much of the efficiency gains through automation. This creates a fundamental conflict: while automated DT generation reduces initial development and updating efforts, it simultaneously increases the complexity of validation and verification, potentially counteracting its intended efficiency gains.

1.3 Objective

This thesis addresses this conflict by developing a data-driven framework for automated VVUQ of automatically generated, simulation-based DTs which have been learned from data. The focus lies on DMFS due to their practical relevance and dynamical, procedural nature. The research can further be specified by the following research questions (RQ):

- **RQ1:** How can automated validation and verification processes for DTs be efficiently implemented to maintain accuracy?
- **RQ2:** Which data-driven approaches are best suited to identify discrepancies between simulated behavior and real operational data in discrete material flow systems?
- **RQ3:** To what extent does the developed framework improve the quality and reliability of DTs compared to traditional VV methods?

This thesis proposes that object-centric event logs—commonly used to generate DTs in manufacturing—can also serve as the foundation for an automated, use-case-independent validation and verification framework. Such an approach would preserve the efficiency benefits of automated generation while ensuring that the resulting DTs meet necessary standards. A key aspect of this approach is the development and monitoring of generic, statistically grounded reference values, which must be quantifiable and have an underlying distribution. The framework will be evaluated using a case study from the discrete material flow domain, providing empirical evidence of its effectiveness in improving model accuracy and efficiency.

1.4 Structure and Methodology

Structure

The thesis is organized into eight chapters. Chapter 2 establishes the theoretical foundation. It begins with broad, domain-specific concepts and progressively narrows the focus to the core topics of this thesis: Automated verification and validation (VVUQ) of simulation-based digital twins (SBDTs) in discrete material flow systems. Section 2.1 introduces material flow planning and simulation, outlining the key elements of production systems, such as processes, resources, and control mechanisms. It also defines key performance indicators (KPIs), essential for evaluating both real and simulated systems, providing the practical context in which DTs operate. Section 2.2 then transitions to the DT concepts. A framework for comparing DTs by Schwede and Fischer (2024) is presented. Special attention is given to data-driven DTs and their subset, automatically generated digital twins (AGDTs). The section concludes by contrasting AGDTs with classical simulation models, highlighting the challenges posed by automatically generated models. Building on this foundation, Section 2.3 presents the principles of process mining (PM) and event log analysis, focusing on object-centric event logs as a data basis for automated validation. This section demonstrates how PM acts as a bridge between real-world process data and model validation, thus enabling continuous verification of SBDTs. Section 2.4 narrows the focus further to VVUQ in the context of SBDTs, beginning with a historical overview of VVUQ methodologies. It then addresses the specific challenges posed by automatically generated models, such as data dependency and lack of transparency in model creation. The section introduces machine learning-based approaches for VVUQ, particularly classification methods for detecting model deviation. It also explores the current state of VVUQ in corporate practice, emphasizing the need for continuous and automated validation processes.

Chapter 3 outlines the methodology for developing the proposed framework. It begins with a requirements analysis, deriving functional, technical, and data format requirements from theoretical findings. The chapter then elaborates on the data-based validation strategy, machine learning-based validation approach, metrics for model evaluation, and online validation with continuous monitoring.

Chapter 4 presents the implementation of the framework, starting with the architecture and system setup, followed by detailed descriptions of event log processing, simulation integration, and the machine learning pipeline.

Chapter 5 presents the case study results, evaluating the framework's effectiveness in improving DT quality. It describes the application scenario and data basis, the automatically generated DT, validation experiments, and result interpretation. It concludes with a comparison to manual validation methods.

Chapter 6 discusses the implications of the results and provides recommendations for future research. It evaluates the framework in light of the research questions, examines the significance of verification in automatically generated DTs, addresses limitations, and explores implications for research and practice.

Chapter 7 concludes the thesis by summarizing key findings and their implications. It addresses the research questions and hypotheses, discusses the significance of the results, acknowledges limitations, and provides recommendations for future work.

Methodology

The thesis follows a Design Science Research approach (DSR). This approach is characterized by the development of artifacts to solve practical problems (Hevner et al., 2004; Peffers et al., 2007). Artifacts in the sense of DSR are created objects or constructs which address the given problem and contribute to both theory and practice. The artifacts are evaluated in a real-world context to demonstrate their effectiveness. The thesis applies the cyclical DSR model, see figure 1.2.

The research paradigm of the thesis is deductive-theory critical (Eberhard, 1987). A conceptual VVUQ framework is developed based on existing theoretical foundations, while deriving new requirements through a requirements analysis. The framework is then applied in a case study to evaluate its effectiveness. The research is critical in that it aims to improve the efficiency and effectiveness of VVUQ for automatically generated DTs. Elements of empirical research are included through the case study and the data-driven approach.

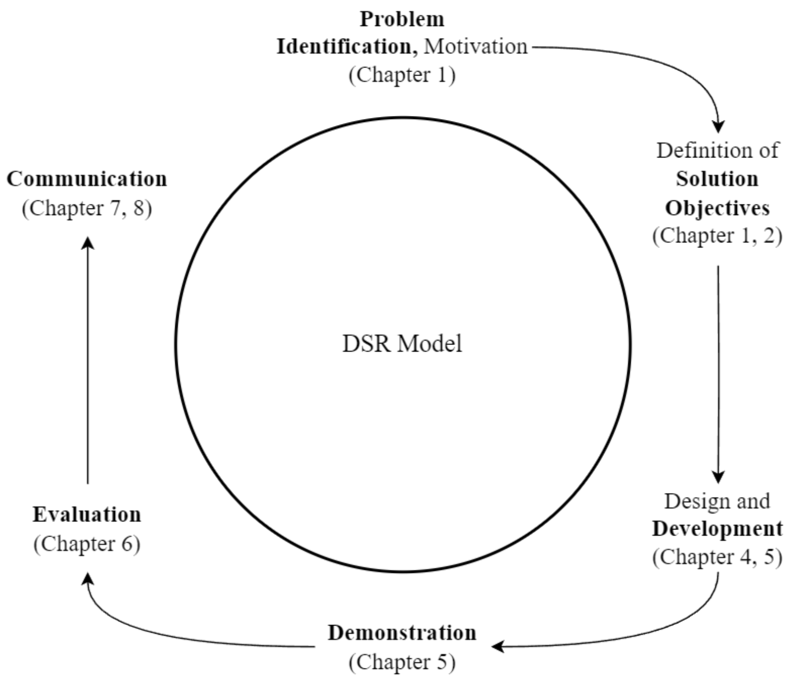

Figure 1.2: The cyclical design science research model. The model consists of six steps. The problem identification (1) refers to the research gap in automated VVUQ of SBDT. Defining the solution objectives (2) specifies the research gap by formulating questions and hypotheses based on the theoretical foundations. The design and development (3) phase includes the development of the framework. The demonstration (4) phase shows the application of the framework in a case study. The evaluation (5) phase assesses the effectiveness of the framework. The communication (6) phase concludes the research by presenting the results.

Chapter 2

Theoretical Foundation

The following chapter provides a theoretical foundation for the research conducted in this thesis. It introduces the basic concepts of material flow planning and simulation, digital twins, process mining, and verification, validation, and uncertainty quantification (VVUQ). The relevance of these concepts in the context of simulation-based digital twins and their application in corporate practice will also be discussed.

2.1 Discrete Material Flow Systems and Simulation

This section begins with an introduction of the underlying concepts of Discrete Material Flow Systems (DMFS) and Simulation Based Digital Twins (SBDT).

2.1.1 Basic Concepts

Discrete material flow systems cannot be fully understood without first clarifying the principles of Discrete Event Simulation (DES) for Discrete Event Systems. In DES, a system changes its state through *events* that occur at specific, discrete time instances; it is assumed that no changes occur between two successive events. Consequently, the state of the system is completely defined by the values of its descriptive variables at each event occurrence (Varga, 2001). The time at which an event occurs is typically marked by a timestamp, and the scientific observation of such systems is conducted by analyzing the discrete *sequence* of events over time (Robinson, 2014).

Simulation, in this context, refers to the process of imitating the operation of a Discrete Event System over time—often through multiple event sequences. This imitation is captured in a model, and the core activities in a simulation involve constructing and experimenting with this model. A high-quality simulation abstracts the essential features of the system, which requires the modeller to have a sound a priori understanding of what “essential” means in the given context. Although the model can later be refined, its quality is primarily measured by its ability to predict outcomes and offer a diverse range of scenarios (Maria, 1997).

In the context of DMFS, their simulation describes the imitation of material flow systems by breaking down continuous flows into discrete events. Such material flow systems can be characterized as “systems processing discrete objects (parts) that move at regular or irregular intervals along transportation routes or conveyor lines, comprising production and logistic systems” (Arnold & Furmans, 2006; Schwede & Fischer, 2024). These systems form the backbone of material flow planning and control structures. The central idea of material flow planning and control is to ensure that material requirements—both in terms of quantity and timing—are met during transportation and storage across the various stages of the supply chain (“Der Materialfluss im Zuliefernetzwerk — integrierte und prozessorientierte Planung und Steuerung”, 2007). Importantly, the time horizon of interest spans from order placement up to delivery. To summarize, DMFS are often simulated using DES, which abstracts the continuous flow of materials into discrete events. The simulation is carried out using a model. The simulation and modeller are embedded in the context of material flow planning and control, which aims to ensure that material requirements are met across the supply chain. Successfully performed material flow planning and control induce high quality data for simulation and modelling purposes.

2.1.2 Comparing DMFS

Because the simulation of DMFS often involves (discrete) event simulation, events in DMFS need to be further differentiated to be comparable. (Arnold & Furmans, 2006) propose to differentiate DMFS into static and dynamic components.

Static components describe the possible states of the system. Possible states can be the set of possible processes given a part or resource, for example. Dynamic components define the concrete material flow for a certain part or order. Static components include parts, resources and processes (Schwede & Fischer, 2024). Parts are transformed by processes using resources sometimes based on orders. Transformation can have an impact on physical properties of the parts (transformation model), spatial position (transition model), the quality of the parts (quality model) and takes time (time model) and uses resources (resource model). Resources have a capacity of handling parts in parallel (resource capacity model) and processes have a predecessor-successors relationship (process model). Dynamic components are used to define the concrete dynamic material flow within the DMFS. There are four components: Order generation, order control, resource control and supply control. Order generation defines the load the system must process. Order control defines how parts are processed, sometimes referred to as routing rules (Milde & Reinhart, 2019). Resource control defines how resources decide to handle processing requests, also sometimes referred to as priority rules. Supply control describes how supply parts are provided (Milde & Reinhart, 2019; Schwede & Fischer, 2024). See the latter source for a more detailed description of the components.

2.1.3 Production Planning and Control

Successful companies use production planning and control frameworks to describe and optimize their DMFS. After establishing a theoretical foundation and simulation approaches for DMFS, this section thus focusses on Production Planning and Control (PPC) as a critical factor influencing the quality and quantity of data generated by Discrete Event Simulation. PPC is the structured approach to planning, scheduling, controlling and managing all aspects of the manufacturing process. It involves the coordination of resources, processes, and orders to meet production goals. PPC is essential for optimizing production processes, reducing costs, and improving quality. The main functions of PPC include production planning, production scheduling, and production control. Production planning involves determining the production capacity, production goals, and production processes. Production scheduling involves creating a detailed schedule for production activities. Production control involves monitoring and controlling production activities to ensure that production goals are met (Kiran, 2019). Scheduling is usually the last step performed before execution of the plan (Pinedo & Pinedo, 2012).

The integration of PPC with simulation models is crucial because it directly affects the data quality used in DES of DMFS. Effective PPC processes anticipate anomalies in the production cycle, allowing for adjustments that maintain system efficiency and reliability. If successful, these adjustments yield high-quality data that enhance the accuracy of simulation outcomes. (Kiran, 2019).

2.1.4 Relevant KPIs and Metrics

Up to this point, DES for SBDT of DMFS has been introduced, outlining the key factors that contribute to a robust simulation. A model differentiation framework proposed by Schwede and Fischer has been briefly presented to facilitate comparison of SBDT. Furthermore, the critical role of PPC in generating high-quality data for simulation has been discussed. These discussions ignored up till now that, even when SBDT are integrated within well-functioning PPC processes, various SBDT models remain prone to errors and inherent trade-offs that must be addressed by the modeller (Tao et al., 2018).

The goal conflict of the modeller when developing SBDT can be described by the following conflict triangle (Balci, 2012; Robinson, 2014):

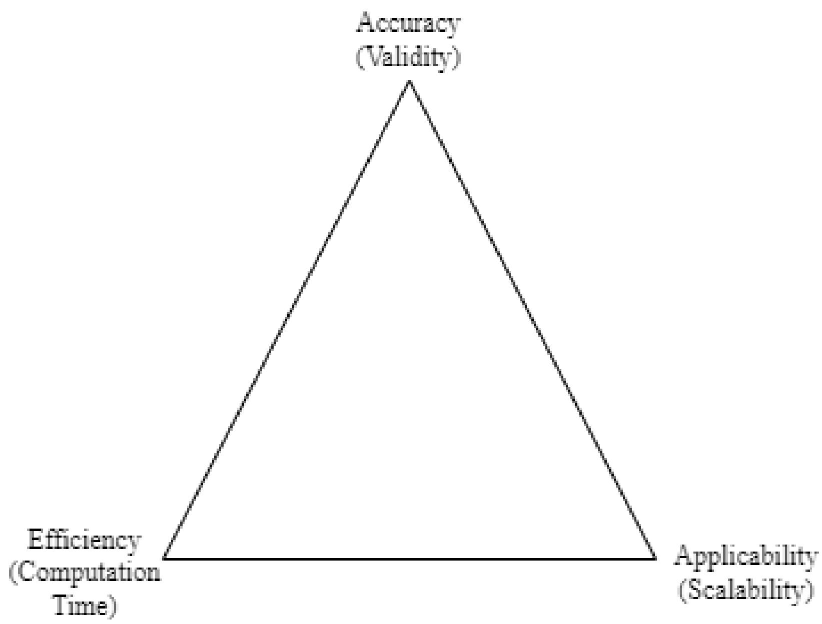

Figure 2.1: The goal conflict of the modeller when developing SBDT. Aiming for higher accuracy (validity) often leads to higher computational costs (efficiency) and reduced scalability (applicability). Reaching more efficiency often leads to reduced accuracy and scalability. Aiming for higher scalability often leads to reduced accuracy and efficiency.

Focusing one of the three dimensions—accuracy (validity), efficiency (computation time), and applicability (scalability)—often leads to trade-offs in the other two dimensions. Oftentimes the data itself is not sufficient to make a decision on which trade-off to make. Limited data points may hinder the modeller from reaching high validity. System architecture may block the system from reaching good scalability. Hardware limitations may hinder the modeller from reaching high efficiency. At other times, corporate management may have a preference for one of the dimensions.

One solution to balance and quantify these goals can be achieved by defining a set of KPIs. Some may already be available through PPC, some may be calculated from DES data or the DES itself. Optimally, the data warehouse provides relevant views (Cui et al., 2020). Because the SBDT in theory mirrors the DMFS, the KPIs gathered from PPC and the DES should yield identical values. Deviations between the KPIs of the SBDT and the DMFS may indicate errors in the SBDT or anomalies in the DFMS. The following KPIs are relevant for the evaluation of SBDT:

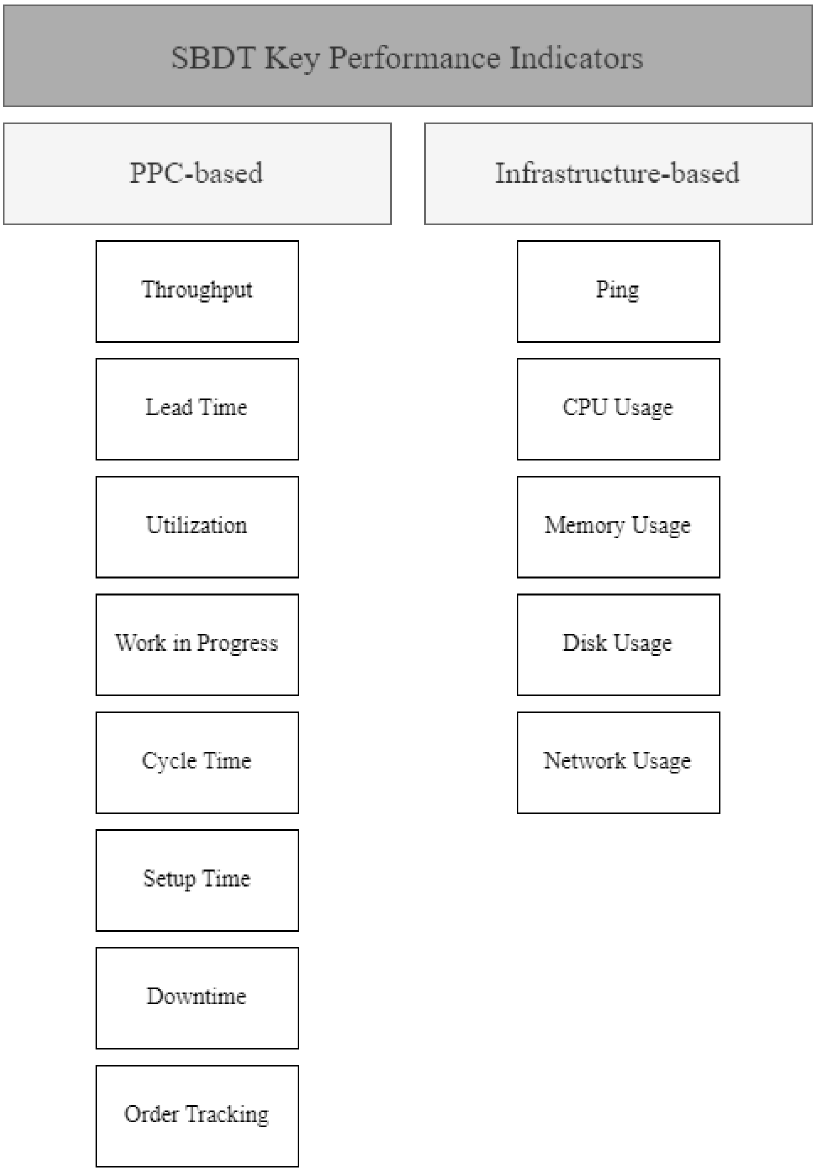

Figure 2.2: SBDT KPIs differentiated by PPC-based and Infrastructure-based indicators.

The PPC related KPIs may be provided by above mentioned data warehouse, because they are highly relevant in the context of production scheduling and -control. Throughput measures the number of produced parts at the last station in a specified period. It is an indicator for the productivity of the manufacturing sytem (Imseitif et al., 2019). Lead time is the cumulative time a part travels through the system from start to sink. It is an indicator for the efficiency of the manufacturing system (Pfeiffer et al., 2016). Cycle time measures the same amount like lead time but focusses only on the active production process, excluding transports and waiting times (Griffin, 1993). Setup time measures the time needed to prepare a machine for a new task. It is an indicator for the flexibility of the manufacturing system (Allahverdi & Soroush, 2008). In the given usecase, we aggregate the setup time for all setup processes. All KPIs presented so far can be calculated dynamically when new data has been sent. Later on, they may serve as an alert system for the modeller to detect deviations between the SBDT and the DMFS, see section 2.4.

The infrastructure related KPIs are derived by sensors from the executing system of the SBDT. Ping time measures the time needed to send a signal from one point to another. It is an indicator for the latency of the infrastructure (Wu et al., 2021). SBDT need to enforce real-time control over the physical entity. The latency thus needs to be as low as possible. In this scenario, one point (sender) is represented by the physical entity and its sensors. The receiving point runs the SBDT. It is advantageous to run the SBDT on Edge to minmize latency- and transmission costs (Li et al., 2018). CPU-, memory-, disk- and network usage metrics are indicators for the load of the infrastructure. They are important to detect bottlenecks in the infrastructure (Li et al., 2018). The first indicator is usually measured in percent of the maximum CPU capacity. The latter three indicators are usually measured in bytes or bits per second (Granelli et al., 2021).

2.2 Digital Twin: Definition and Concepts

The latter section gave a short intrduction into DFMS, DES, its metrics and the corporate processes accompanying the SBDT. Now, we shed light on the DT itself. For a short introduction to the topic, see chapter 1.

Like introduced in the predecessing chapter, DT inherent the highest order of modelling fidelity compared to DM or DS. There are different definitions of DT present in the literature (Boschert & Rosen, 2016; Demkovich et al., 2018; Glaessgen & Stargel, 2012; Grieves, 2014; Kritzinger et al., 2018; Negri et al., 2017; Tao et al., 2018; Zehnder & Riemer, 2018; Zheng et al., 2019). Each of them highlights different aspects of the DT. This thesis uliziles the definition by (Grieves, 2014) which highlights the conceptual elements of the twin and its lifecylce focus:

> The digital twin concept (...) contains three main parts: A) Physical products in real space, (B) virtual products in virtual space and (C) the two-way connections of data and information that tie the virtual and real products together. (Grieves, 2014)

The physical product is the entity which will be modelled. The virtual product is the DT itself, but also its infrastructure, for example data services making the real-time data flow possible (Tao et al., 2018). The two-way connection is the data flow between the physical and the virtual product. The data flow is bidirectional. Zehnder and Riemer add that the data flow may contain meta data ”describing the data source and its context”. Also the connection protocol is of importance here (e.g. MQTT or REST). TCP may be the method of choice as it ensures that the packages arrive in the correct order and without errors (Li et al., 2018).

2.2.1 Types of Digital Twins

Now that a unified understanding of DT has been established, this section focuses on how DT may be learned from different sources of information. The following list includes the most relevant types of DT with a focus on different kinds of information sources:

- Simulation-based DT (SBDT) (Lugaresi & Matta, 2021; Martinez et al., 2018)
- Data-driven DT (DDDT) (Friederich et al., 2022; He et al., 2019)
- Hybrid Digital Twins (HDT) (Huang et al., 2023; Luo et al., 2020)

SBDTs (Boschert & Rosen, 2016; Lugaresi & Matta, 2021; Martinez et al., 2018) are based on DES. They utilize discrete event simulation (see section 2.1) to create a dynamic representation of the physical system (Pantelides & Renfro, 2013; Schluse & Rossmann, 2016). To incorporate a SBDT into workflows and processes, suitable data structures must be in place beforehand (Boschert & Rosen, 2016). DES may improve the predictive capabilities of the model compared to manual twin creation. DES is able to model causal relationships between the events (Francis et al., 2021). In contrast, the development of a realistic simulation model requires experts and time (Charpentier & Véjar, 2014). If the simulation model fails to capture recent behaviour of the physical entity, a recalibration is mandatory (Friederich et al., 2022). In a nutshell, SBDTs are a step forward to speed up the creation and updating processes of DTs.

DDDT rely on the utilization of data to model the physical entity. The data may be gathered from sensors, data warehouses or other sources. The data is used to train a model which represents the physical entity. The model may be a neural network, a decision tree or another machine learning model. The model is then used to predict future states of the physical entity. The model may be updated with new data to increase its accuracy (Friederich et al., 2022; He et al., 2019). For a more detailed description of DDDT including its up- and downsides, see subsection 2.2.2.

HDT combine different sources of information to create a more accurate model of the physical entity. The sources may be simulation models (see section 2.1), data-driven models (see subsection 2.2.2) or physics-based models. Physics-based models contain information about the physical properties and behaviors of the entity. They do not have to learn these characteristics from the data because this information is made available to the model *a priori* (Aivaliotis et al., 2019; Kapteyn et al., 2022). The simulation based models accompanying the physics-based one obeys characteristics of SBDT, see above. The combination of different sources may make the HDT more robust and a faster learner. HDT unite the advantages of SBDT with the knowledge advantage physics based models have. Unfortunately, they also inherit the disadvatages of SBDT. Physics-based models may also involve heavy involve heavy computational costs and domain expertise (Kapteyn et al., 2022).

2.2.2 Data-Driven Digital Twins

While SBDTs and HDT possess not negligible computational costs and require domain expertise, DDDT are able to learn from data without the need for a hand-written simulation model. The DDDT *learns* the model. Learning in the context of DDDT is not trivial, several approaches have been proposed in the literature (Francis et al., 2021; Friederich et al., 2022; He et al., 2019). Often times Data Science methods come to work. The learning process may be supervised or unsupervised. Supervised learning uses labeled data to train the model (Cunningham et al., 2008). The label can symbolize different values of interest. Unsupervised learning uses unlabeled data to train the model (Barlow, 1989). Often times, the task at hand is to group the data into different categories. Data sources of interest may be process data, resource data or order data (Biesinger et al., 2019). The learning process may be online or offline. Offline learning uses the data *once* for training, validation and testing, while online learning continuously updates the model with new data to adapt to changes in the physical system. Online learning is thus able to capture new trends in the data and to foresee concept drift (Tsymbal, 2004). DDDT have to be differentiated from data-driven simulation (Charpentier & Véjar, 2014), which involves human invervention to create highly individual solutions for the physical entity. The key difference is that every characteristic has to be explicitly described in the model by the expert, there are no efforts to let an intelligent algorithm learn these by itself. DDDT thus rely less on domain expertise and manual model creation. A suitable model may be able to capture relevant trends in the data and to predict outcomes which describe most of the characteristics of the physical entity. (Francis et al., 2021) propose several key elements a DDDT must contain to be termed *data-driven*:

- **Data Collection:** The relevant entities to be modelled have to be identified. This activity involves data gathering of the identified entities and ensuring a steady data stream to a database. The data may be gathered from sensors, data warehouses or other sources.

- **Data Validation:** This step involves cleaning and preprocessing the data. The data may contain missing values, outliers or other errors. The data has to be cleaned and preprocessed to ensure a high quality of the model. Plausibility checks may be performed to ensure the data is correct.

- **Knowledge Extraction:** After the data has been collected and cleaned, events have to be detected. Francis et al. utilize process mining terms in this context, such as evemt detection and process discovery. The main goal in this step is to find a common ground on which events are of interest. The thesis later dives deeper into Process Mining techniques applied here, see section 2.3.

- **(Semi-)automatic Simulation Modeling:** The data is used to train a model. This step may use offline or online data as a stream. The model may be a neural network, a decision tree or another machine learning model. The model is then used to predict future states of the physical entity. The model may be updated with new data to increase its accuracy.

- **Continuous Model Validation:** Interestingly,Francis et al. propose a continuous model validation. In the online learning case, they recommend to use the steady data stream to apply validation techniques continuously, see section 2.4 The validation may be performed by comparing the model predictions with the real data. If the model deviates from the real data, the model may be recalibrated.

DDDTs go one step further than SBDT and minimize the influence of the human in the loop (Francis et al., 2021; Friederich et al., 2022). Faster model development and updating activities are the result. The third reason to automate DT endevours elaborated by Schwede and Fischer, increasing prediction quality, rises and falls with the data quality, thus the gathering and preprocessing efforts of the modeller. Extrinsic factors like the number of data points available also play into the equation (curse of dimensionality). DDDT should avoid biased or noisy predictions at all costs. The identification of *relevant* events poses the risk of introducing a selection bias, rather a confirmation bias. The modeller may have the tendency to select events which confirm his or her hypothesis. Ramdom sampling may be a solution to this problem, but can destroy sequential information patterns in event sequences. Overall DDDT are a promising approach to model the physical entity. If the right balance between human involvement and automated learning is found, it may be an efficient solution (Francis et al., 2021). Thinking one step ahead, employing data-based VVUQ approaches may also be a gamechanger. This topic will be discussed in Section section 3.3.

2.2.3 Automatically Generated Digital Twins

AGDT mitigate the slight inefficiencies of DDDT by eliminating the human in the loop. The framework by Francis et al. already discernibled automatically generated DT.

2.2.4 Definitions and Differences from Classical Simulation Literature

2.3 Process Mining and Event Logs

2.3.1 Core Concepts

2.3.2 Object-Centric Event Logs as a Data Basis

2.3.3 Process Mining as Enabling Technology

2.4 VVUQ in the Context of Simulation-Based Digital Twins

2.4.1 Historical Development of VVUQ Concepts

2.4.2 Requirements of VVUQ for Automatically Generated Models

2.4.3 Theoretical Argumentation for Merging Verification, Validation and Uncertainty Quantification

2.4.4 Machine Learning-Based Approaches

2.4.5 VVUQ in the Context of Digital Twins

2.4.6 VVUQ in Corporate Practice

Chapter 3

Framework Design

Chapter 4

Implementation

Chapter 5

Testing

Chapter 6

Discussion

6.1 Evaluation of the Developed Framework

6.2 Significance of Verification in Automatically Generated Digital Twins

6.3 Limitations of Automated Validation

6.4 Implications for Research and Practice

Chapter 7

Conclusion and Outlook

7.1 Summary of the Key Findings

7.2 Methodological and Theoretical Insights

7.3 Outlook

7.4 Recommendations for Practical Application

Bibliography

Abdoune, F., Nouiri, M., Cardin, O., & Castagna, P. (2023). Digital twin lifecycle: Core challenges and open issues. In T. Borangiu, D. Trentesaux, & P. Leitão (Eds.), *Service oriented, holonic and multi-agent manufacturing systems for industry of the future: Proceedings of sohoma 2022* (pp. 157–167). Springer International Publishing. https://doi.org/10.1007/978-3-031-19647-3_14

Aivaliotis, P., Georgoulias, K., Arkouli, Z., & Makris, S. (2019). Methodology for enabling digital twin using advanced physics-based modelling in predictive maintenance. *Procedia Cirp*, *81*, 417–422.

Allahverdi, A., & Soroush, H. M. (2008). The significance of reducing setup times/setup costs. *European journal of operational research*, *187*(3), 978–984.

Arnold, D., & Furmans, K. (2005). *Materialfluss in logistiksystemen* (4th ed.). Springer-Verlag. https://doi.org/10.1007/b139029

Arnold, D., & Furmans, K. (Eds.). (2006). *Materialfluss in logistiksystemen* (5th ed.). Springer Berlin Heidelberg.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, *58*, 82–115.

Balci, O. (2012). A life cycle for modeling and simulation. *Simulation*, *88*(7), 870–883.

Barlow, H. B. (1989). Unsupervised learning. *Neural computation*, *1*(3), 295–311.

Biesinger, F., Meike, D., Kraß, B., & Weyrich, M. (2019). A digital twin for production planning based on cyber-physical systems: A case study for a cyber-physical system-based creation of a digital twin. *Procedia CIRP*, *79*, 355–360. https://doi.org/10.1016/j.procir.2019.02.092

Bitencourt, L., Silva, M., & Costa, J. (2023). Building trust in digital twin through verification and validation. *Proceedings of the 2023 International Conference on Digital Twin Technologies*, 55–63. https://doi.org/10.1000/bitencourt.2023.001

Boschert, S., & Rosen, R. (2016). Digital twin—the simulation aspect. *Mechatronic futures: Challenges and solutions for mechatronic systems and their designers*, 59–74.

Charpentier, P., & Véjar, A. (2014). From spatio-temporal data to manufacturing system model. *Journal of Control, Automation and Electrical Systems*, *25*(5), 557–565. https://doi.org/10.1007/s40313-014-0111-2

Cui, Y., Kara, S., & Chan, K. C. (2020). Manufacturing big data ecosystem: A systematic literature review. *Robotics and computer-integrated Manufacturing*, *62*, 101861.

Cunningham, P., Cord, M., & Delany, S. J. (2008). Supervised learning. In *Machine learning techniques for multimedia: Case studies on organization and retrieval* (pp. 21–49). Springer.

Demkovich, N., Yablochnikov, E., & Abaev, G. (2018). *Multiscale modeling and simulation for industrial cyber-physical systems*. https://doi.org/10.1109/ICPHYS.2018.8387674

Der materialfluss im zuliefernetzwerk — integrierte und prozessorientierte planung und steuerung. (2007). In *Logistik in der automobilindustrie: Innovatives supply chain management für wettbewerbsfähige zulieferstrukturen* (pp. 73–100). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-68114-4_4

Eberhard, K. (1987). *Einführung in die erkenntnis-und wissenschaftstheorie: Geschichte und praxis der konkurrierenden erkenntniswege*. Kohlhammer.

Francis, D. P., Lazarova-Molnar, S., & Mohamed, N. (2021). Towards data-driven digital twins for smart manufacturing. *Proceedings of the 27th International Conference on Systems Engineering, ICSEng 2020*, 445–454.

Frank, A. G., Dalenogare, L. S., & Ayala, N. F. (2019). Industry 4.0 technologies: Implementation patterns in manufacturing companies. *International Journal of Production Economics*, *210*, 15–26. https://doi.org/10.1016/j.ijpe.2019.01.004

Friederich, J., Francis, D. P., Lazarova-Molnar, S., & Mohamed, N. (2022). A framework for data-driven digital twins of smart manufacturing systems. *Computers in Industry*, *136*, 103586. https://doi.org/10.1016/j.compind.2022.103586

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*(1), 1–58. https://doi.org/10.1162/neco.1992.4.1.1

Glaessgen, E., & Stargel, D. (2012). The digital twin paradigm for future NASA and U.S. air force vehicles. *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*. https://doi.org/10.2514/6.2012-1818

Granelli, F., Capraro, R., Lorandi, M., & Casari, P. (2021). Evaluating a digital twin of an iot resource slice: An emulation study using the eliot platform. *IEEE Networking Letters*, *3*(3), 147–151.

Grieves, M. (2014). *Digital twin: Manufacturing excellence through virtual factory replication* (tech. rep.) (White Paper).

Griffin, A. (1993). Metrics for measuring product development cycle time. *Journal of Product Innovation Management: An International Publication of the Product Development & Management Association*, *10*(2), 112–125.

He, R., Chen, G., Dong, C., Sun, S., & Shen, X. (2019). Data-driven digital twin technology for optimized control in process systems. *ISA transactions*, *95*, 221–234.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75–105.

Hua, E. Y., Lazarova-Molnar, S., & Francis, D. P. (2022). Validation of digital twins: Challenges and opportunities. *2022 Winter Simulation Conference (WSC)*, 2900–2911. https://doi.org/10.1109/WSC57314.2022.10015420

Huang, Z., Fey, M., Liu, C., Beysel, E., Xu, X., & Brecher, C. (2023). Hybrid learning-based digital twin for manufacturing process: Modeling framework and implementation. *Robotics and Computer-Integrated Manufacturing*, *82*, 102545.

Imseitif, J., Tang, H., & Smith, M. (2019). Throughput analysis of manufacturing systems with buffers considering reliability and cycle time using des and doe. *Procedia Manufacturing*, *39*, 814–823.

Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the digital twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, *29*, 36–52. https://doi.org/10.1016/j.cirpj.2020.02.002

Judijanto, L., Qadriah, L., Prabowo, I. A., Widyatmoko, W., & Sabila, P. C. (2024). Trends in digital twin technology for industry 4.0: A bibliometric study. *The Eastasouth Journal of Information System and Computer Science*, *2*(02), 92–104.

Kapteyn, M. G., Knezevic, D. J., Huynh, D., Tran, M., & Willcox, K. E. (2022). Data-driven physics-based digital twins via a library of component-based reduced-order models. *International Journal for Numerical Methods in Engineering*, *123*(13), 2986–3003.

Kiran, D. (2019). *Production planning and control: A comprehensive approach*. Butterworth-heinemann.

Kovacs, G., & Kostal, P. (2016). Mathematical description of material flow. *Materials science and technology*, *1*.

Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W. (2018). Digital twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, *51*(11), 1016–1022. https://doi.org/10.1016/j.ifacol.2018.08.474

Li, H., Ota, K., & Dong, M. (2018). Learning iot in edge: Deep learning for the internet of things with edge computing. *IEEE network*, *32*(1), 96–101.

Lim, K. Y. H., Zheng, P., & Chen, C.-H. (2020). A state-of-the-art survey of digital twin: Techniques, engineering product lifecycle management and business innovation perspectives. *Journal of Intelligent Manufacturing*, *31*(6), 1313–1337.

Lugaresi, G., & Matta, A. (2021). Automated digital twins generation for manufacturing systems: A case study. *IFAC-PapersOnLine*, *54*(1), 749–754. https://doi.org/10.1016/j.ifacol.2021.08.087

Luo, W., Hu, T., Ye, Y., Zhang, C., & Wei, Y. (2020). A hybrid predictive maintenance approach for cnc machine tool driven by digital twin. *Robotics and Computer-Integrated Manufacturing*, *65*, 101974.

Maria, A. (1997). Introduction to modeling and simulation. *Proceedings of the 29th conference on Winter simulation*, 7–13.

Martinez, G. S., Sierla, S., Karhela, T., & Vyatkin, V. (2018). Automatic generation of a simulation-based digital twin of an industrial process plant. *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, 3084–3089.

Milde, M., & Reinhart, G. (2019). Automated model development and parametrization of material flow simulations. *2019 Winter Simulation Conference (WSC)*, 2166–2177. https://doi.org/10.1109/WSC40007.2019.9004702

Negri, E., Fumagalli, L., & Macchi, M. (2017). A review of the roles of digital twin in cps-based production systems. *Procedia Manufacturing*, *11*, 939–948. https://doi.org/10.1016/j.promfg.2017.07.198

Oztemel, E., & Gursev, S. (2020). Literature review of industry 4.0 and related technologies. *Journal of intelligent manufacturing*, *31*(1), 127–182.

Pantelides, C. C., & Renfro, J. G. (2013). The online use of first-principles models in process operations: Review, current status and future needs. *Computers & Chemical Engineering*, *51*, 136–148.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, *24*(3), 45–77.

Pfeiffer, A., Gyulai, D., Kádár, B., & Monostori, L. (2016). Manufacturing lead time estimation with the combination of simulation and statistical learning methods. *Procedia Cirp*, *41*, 75–80.

Pinedo, M. L., & Pinedo, M. L. (2012). Design and implementation of scheduling systems: Basic concepts. *Scheduling: theory, algorithms, and systems*, 459–483.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Robinson, S. (2014). *Simulation: The practice of model development and use*. Bloomsbury Publishing.

Schluse, M., & Rossmann, J. (2016). From simulation to experimentable digital twins: Simulation-based development and operation of complex technical systems. *2016 IEEE international symposium on systems engineering (ISSE)*, 1–6.

Schwede, C., & Fischer, D. (2024). Learning simulation-based digital twins for discrete material flow systems: A review. *2024 Winter Simulation Conference (WSC)*, 3070–3081.

Sel, K., Hawkins-Daarud, A., Chaudhuri, A., Osman, D., Bahai, A., Paydarfar, D., Willcox, K., Chung, C., & Jafari, R. (2025). Survey and perspective on verification, validation, and uncertainty quantification of digital twins for precision medicine. *npj Digital Medicine*, *8*(1), 40.

Sepasgozar, S. M. (2021). Differentiating digital twin from digital shadow: Elucidating a paradigm shift to expedite a smart, sustainable built environment. *Buildings*, *11*(4), 151.

Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., & Sui, F. (2018). Digital twin-driven product design, manufacturing and service with big data. *The International Journal of Advanced Manufacturing Technology*, *94*(9-12), 3563–3576. https://doi.org/10.1007/s00170-017-0233-1

Thiede, S., Seow, Y., Andersson, J., & Johansson, B. (2013). Environmental aspects in manufacturing system modelling and simulation—state of the art and research perspectives. *CIRP Journal of Manufacturing Science and Technology*, *6*(1), 78–87. https://doi.org/10.1016/j.cirpj.2012.10.002

Trauer, J., Schweigert-Recksiek, S., Schenk, T., Baudisch, T., Mörtl, M., & Zimmermann, M. (2022). A digital twin trust framework for industrial application. *Proceedings of the Design Society*, *2*, 293–302.

Tsymbal, A. (2004). The problem of concept drift: Definitions and related work. *Computer Science Department, Trinity College Dublin*, *106*(2), 58.

Varga, A. (2001). Discrete event simulation system. *Proc. of the European Simulation Multiconference (ESM'2001)*, *17*.

Wu, Y., Zhang, K., & Zhang, Y. (2021). Digital twin networks: A survey. *IEEE Internet of Things Journal*, *8*(18), 13789–13804.

Zehnder, P., & Riemer, D. (2018). Representing industrial data streams in digital twins using semantic labeling. *2018 IEEE International Conference on Big Data (Big Data)*, 4223–4226.

Zhang, L., Zhou, L., & Horn, B. K. (2021). Building a right digital twin with model engineering. *Journal of Manufacturing Systems*, *59*, 151–164. https://doi.org/10.1016/j.jmsy.2021.02.009

Zhao, Y. F., Xie, J., & Sun, L. (2024). On the data quality and imbalance in machine learning-based design and manufacturing—a systematic review. *Engineering*.

Zheng, Y., Yang, S., & Cheng, H. (2019). An application framework of digital twin and its case study. *Journal of ambient intelligence and humanized computing*, *10*, 1141–1153.