Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 104 (2021) 98–103

www.elsevier.com/locate/procedia

54th CIRP Conference on Manufacturing Systems

Continuous adaption through real data analysis turn simulation models into digital twins

Leonard Overbeck[a,*], Oliver Brützel[a], Michael Teufel[a], Nicole Stricker[a], Andreas Kuhnle[a], Gisela Lanza[a]

[a]wbk Institute of Production Science, Karlsruhe Institute of Technology, Kaiserstr. 12, 76131 Karlsruhe, Germany

* Corresponding author. Tel.: 0049 1523 9502641; *E-mail address:* leonard.overbeck@kit.edu

Abstract

Digital twins of production systems enable new forms of production control, flexibility and continuous improvement. While off-the-shelf software for discrete-event simulation permits the fast implementation of rough simulation models with sufficient accuracy for project-based analysis, they lack the precision and generality of a digital twin. This paper presents an approach to close the gap between model and reality by continuous and iterative updates enabled by connecting the simulation model to IT systems and smart data analysis. However, handling different databases requires a generative and flexible modelling approach as well as suitable algorithms for probability distribution estimation and control logic identification. The presented approach was validated at a real world example from the automotive industry where an average deviation of output to reality per week of 0.1% was achieved, proving the effectiveness of the approach.

Keywords: Digital Twin; Discrete-event simulation; Data analysis; Contiunuous improvement process; Material Flow; Process Mining; Flexibility

1. Introduction

Because of accelerating advances in science and technology and fast changing customer preferences and requirements, the life cycle of many products continuously decreases. Companies hence have to constantly adapt their production systems to new product variants [1]. Maintaining productivity also for variants that the system was not originally designed for and managing the transition between product generations as smooth as possible requires a constant analysis of the system, because the insights become obsolete rapidly.

An important tool for the analysis and improvement of dynamic systems is simulation, in the case of production systems mostly discrete-event material flow simulation is used. The disadvantage of simulation is the high manual effort required for modelling, data acquisition and preparation. The traditional approach to simulation is project based [2] so that the model is used only for a given set of tasks in a predefined timeframe and abandoned afterwards.

The goal of the proposed approach is to extend the usage time and improve the accuracy of a discrete-event simulation model while maintaining or even reducing the initial manual effort for modelling and implementation. Increasing benefits by constant costs will improve the return on investment of simulation in production.

To achieve this, a methodology for regular updates and constant improvement of an existing simulation model was developed, which turns a manually created simulation model into a real digital twin of the production system. The automated simulation data update is explained for two exemplary parameters and validated on an automotive industry use case.

The paper is divided into the following parts: After the introduction, Chapter 2 presents previous works and derives the open research question. Chapter 3 describes the methodology

10.1016/j.procir.2021.11.017

how to turn a simulation model into a digital twin of the production system. The use case is presented in Chapter 4 and the validation of two exemplary parameters is shown in Chapter 5. Finally, the paper is summarized, and an outlook provided.

2. Literature review

[3] describe the digital twin concept as the next wave in simulation, which turns simulation into a core functionality of the system by complete integration of the simulation model into the product-service-system and its assistance along its entire life cycle.

[4] define a digital twin as following: "*A Digital Twin (DT) is more than a simple model or simulation. A DT is a living, intelligent and evolving model, being the virtual counterpart of a physical entity or process. It follows the lifecycle of its physical twin to monitor, control, and optimize its processes and functions.*" A key feature of digital twins must therefore be their ability to automatically adapt to changes in the real system through real data.

The problem of the short usage period of most simulation models in which a lot of effort was put in, is a regularly discussed subject in research communities which use and enhance simulation as a tool.

One of the first papers covering this subject remains on a theoretical level and dismembers the reusable system in its components [5]. Many of the discussed issues as the reuse of code have in the meantime already been solved i.e. by object-oriented programming.

[6] calls the full model reuse the "holy grail" of some parts of the simulation community. One answer is presented by [7], who develops an approach for the automated generation of adaptive simulation models based on a standardized data framework called Core Manufacturing Simulation Data. This adaptive simulation model approach includes dynamic behavior i.e. priority rules for lot sizing, resource occupation and production order. This approach requires the data to be provided according to this standard, which is an obstacle for its industrial application.

The same downside has the work of [8] which also relies on a certain data model described with an ontology. The authors give a formal description framework which enables the automatic creation of simulation models.

[9] call the same concept "after-use" and distinguish between re-use and further-use, each bringing different benefits but also obstacles.

A theoretic discussion on several topics concerning the reuse of simulation components ranging from small portions of code to full models is given by [10].

[11] define different ways how real input data can be transferred into the simulation model. The approach presented in this paper follows the methodology, in which external data is automatically populated, because of the centralized data acquisition, high accessibility of data and the easy transfer of collected input data to other simulation tools.

The concept of self-adaptability of discrete-event simulation (SADES) is discussed by [12]. A feedback structure which should enable adaption is proposed but stays on a theoretic level without prototype implementation.

Despite many approaches, the idea of creating a truly adaptive simulation remains an open challenge to research. Especially the issue of automated modelling of the dynamic behavior (e.g. material flow) remains unsolved. Early attempts were made by [13] using a meta-model for the description of strategies giving them a predefined structure. The emulation of scheduling strategies with the help of machine learning is tested by [14].

Another approach to the automated simulation of dynamic behavior can be the use of process mining. In order to examine the behaviour of real production processes, [15] develops a system architecture for the analysis of manufacturing data using process mining techniques. Furthermore, [16] presents an approach to automatically construct simulation models as coloured petri nets based on process execution logs. The relation of discrete-event simulation and process mining is discussed by [17]. Process mining could be used to create fact-

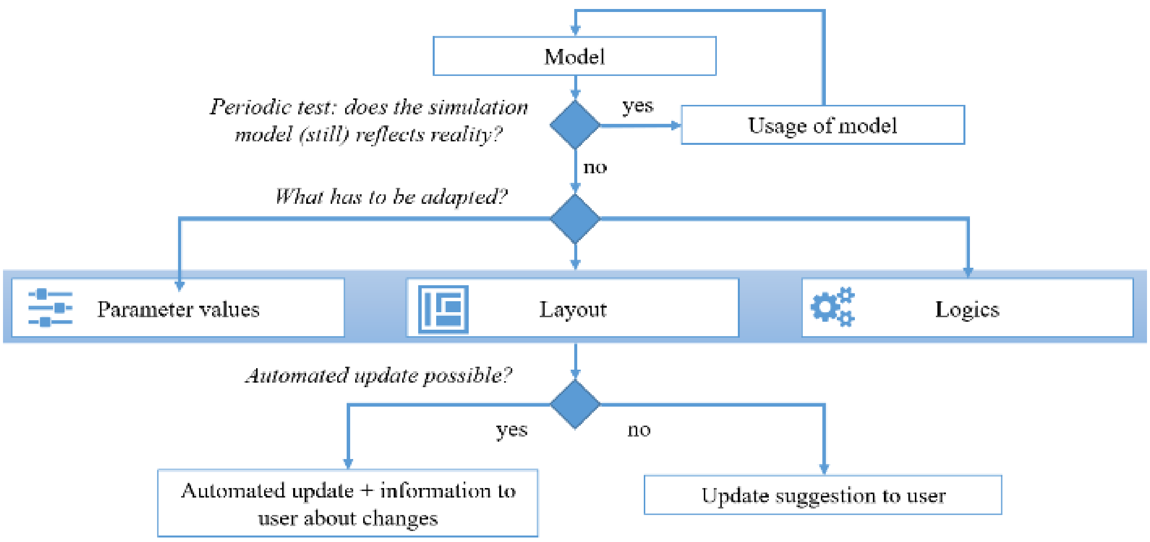

Fig. 1. Update logic of Digital Twin

based simulation models and automate parts of the modelling process. A real-world application of process mining for simulation creation cannot yet be found.

Several authors discuss the application of digital twins in manufacturing but do not discuss the data integration and information extraction needed for the digital twin. Extensive reviews of the numerous applications of digital twins in manufacturing can be found in [18] and [19].

The open research question that this work wants to address is therefore the development and implementation of a general concept of how a common simulation model can be turned into a digital twin of the production system. Particular focus lies on the automated capture of dynamic behaviour using process mining.

3. Own Approach

3.1. Methodology

To increase validity and prolong usability a simulation model has to be frequently compared to reality and adapted if needed. This task is highly repetitive and should thus be automated, which in turn leads to convergence of the simulation model to reality turning it to a real digital twin.

Fig. 1 explains how a simulation model of a production system is turned into a digital twin of the production system by periodic comparison with reality and, if necessary, appropriate updates. Starting point is a manually defined and implemented initial rough simulation model of the production system using predefined generic building blocks, for example as defined in [20] for semi-automated production systems.

Following the digital twin definition of [21], the initial rough model would resemble the digital master of the production system, which depicts the underlying abstract concept of the system, and the collected input data resembles the digital shadow of the system, which incorporates all relevant data of a specific instance of the system. This analogy holds true only partially since the model itself can be adapted, but it gives a good guideline in understanding our approach.

The validation pipeline is described in Fig. 2. After the definition of an appropriate time period that shall be considered for validation, the relevant data is queried from the data bases. This includes produced parts in this period as well as extraordinary events that might have to be considered explicitly in the validation runs. The number of simulation runs depends on the inherent volatility of the system and a trade-off between computation time and statistical significance has to be made. After step 5, the execution of the simulation runs, the metrics to compare simulation and reality have to be calculated.

If the difference between simulation model and reality exceeds a certain threshold, the model has to be adapted until the difference to reality is acceptable. Difference between model and reality can be measured by the output in a given time frame or more specific by the comparison of certain key values of the system, for example buffer occupation or machine workload. The possible adaptions of the model can be divided into three categories: parameter values, layout, and logics. The adaption of each of these categories require different data sources, validation and adaption algorithms. Table 1 lists examples for the contents of each category. Since a detailed discussion of each component would exceed the paper limits, three examples will be discussed in the following in greater detail to give insights in the comparison and adaption algorithms. The representatives for parameter values will be machine processing times and scrap rate and for logics it will be the material flow.

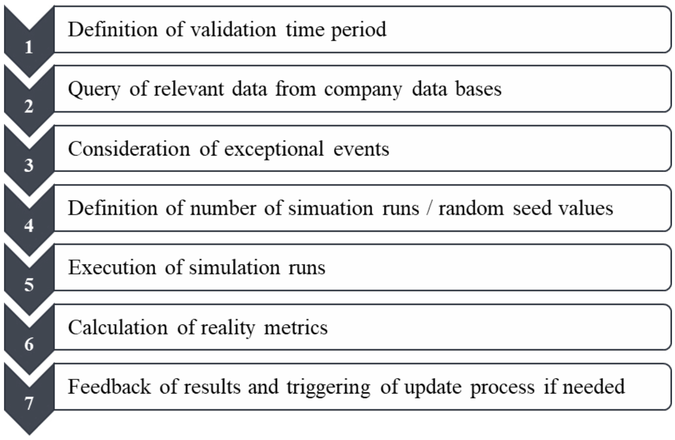

Fig. 2. Validation pipeline of digital twin

Table 1. Extract of components of discrete-event material flow simulation

Parameter values	Layout	Logics
Machine processing times	Number and position of machines	Material flow
Manual processing times	Number and position of conveyors	Scheduling
Transport time	Number and positions of buffers	Worker control
Scrap rate	Number of workers	Maintenance planning
Failure behavior	…	Replenishment method
Buffer sizes	…	…

3.2. Machine processing time

Machining time is a crucial constraint in most production systems. Processing times are normally captured by the machines, but the raw production data often has to be preprocessed and analyzed to be usable, since it can be disturbed by external influences as failures, human worker interaction or commissioning. For simulation, not only the average value of a parameter is important but also the underlying distribution. This requires besides the mean value also at least the standard deviation, which should be calculated from historic data.

The probability density functions for the machine processing times are obtained according to the steps of [22], but instead of a χ^2-test an Anderson-Darling-test is used to check the fit of the calculated distribution with the historic data, since this test shows better performance. If the statistical hypothesis test fails, the data point furthest away from the calculated distribution mean is deleted. Then a new normal distribution is calculated based on the reduced data set and tested again. This process is repeated automatically until the distribution matches with the remaining historic data, which in this way is cleaned of outliers. An exemplary comparison of historic data (in blue) to the calculated distribution is shown in Fig. 3.

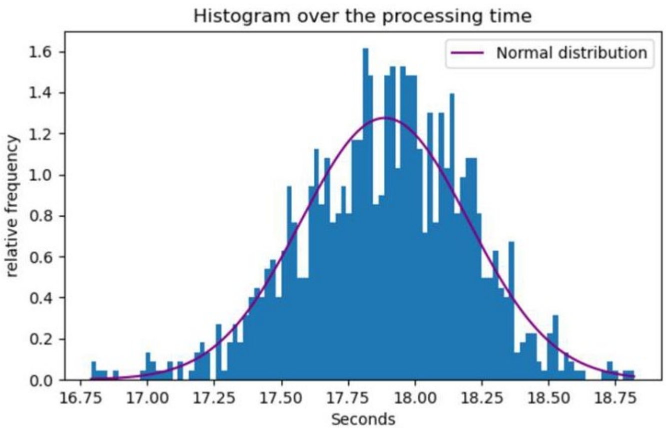

Fig. 3. Historic data and calculated normal distribution

3.3. Automatic input of scrap rate

The scrap rate is another important parameter for modelling the production system. It can be calculated for example using the quality protocol of testing stations or using the difference of products started to products finished successfully considering the products still in production. The general formula is:

$$scrap\ rate = \frac{\#\ parts\ failing\ test}{\#\ parts\ tested} \qquad (1)$$

3.4. Deduction of material flow

After all machines as the core objects of production are initialized their connections must be modelled. The key element, which connects machines to one another, is the flow of parts through the system. It varies for different product variants, as they require different processing steps. Furthermore, parallel machines may exist. In addition, rework stations may be needed if quality problems appear.

Table 2. Example of an event log for production data

Case id	Event id	Attributes				
(Part id)		**Timestamp**	**Activity**	**Variant**	**Result**	**…**
103022	30010	15-06-2020:09:32	St. 3.1.1	887112	good	…
103022	30011	15-06-2020:10:12	St. 3.1.2	887112	good	…
103022	30014	15-06-2009:11:04	St. 7.1.1	887112	good	…
…	…	…	…	…	…	…
103045	30078	16-06-2020:11:01	St. 3.1.1	887112	good	…
103045	30079	16-06-2020:11:47	St. 3.1.2	887112	good	…
103045	30080	16-06-2020:12:13	St. 5.1.1	887112	good	…
…	…	…	…	…	…	…

A simulation model should be able to describe different flow behaviours. This could be done using planning data. However, in the context of the digital twin, it is important to model the real behaviour and therefore use real production data. Event logs from Manufacturing Execution Systems (MES) as shown in Table 2 can be used for process analysis. For a specific part the process at each workstation is documented with timestamps, result status and other information such as the product variant.

Process mining methods can be applied to the event log if enough process instances are available and identifiable, in this example as *part id*. Furthermore, it is necessary to sort the activities according to the timestamps.

To derive the material flow the Alpha Algorithm [23] was applied. This process discovery method examines the order of activities in the different sequences. It results in relations of activities and control flow patterns and creates a Petri net as shown in Fig. 4. Parallel paths and decision points are modelled.

Satisfying results require an appropriate data selection and preparation. For example, it is advisable to filter the event log for a product variant and recognize the specific material flow for this variant. To maintain the general process, rare or unwanted process characteristics have to be filtered out. For example, only process instances that did not require rework shall be considered. This would exclude the path via the rework stations. If data is queried over a fixed period, incomplete sequences may be included. Rare or incomplete process variants can be removed, or processes can be filtered by the number of activities they contain. If certain start or end activities are required, this can also be considered in the preparation.

Based on the discovered process models the objects of a simulation model are connected. In addition, routing tables are created for product variants, which are used in the model. Generated parts receive these tables depending on their product variant. To ensure routing tables remain valid, the derivation of the material flow has to be repeated regularly. Therefore, the described approach should be automated via a script, which executes the steps. New required routing tables are created automatically, and existing tables can be checked for correctness. A direct connection to the production database is necessary. Relevant data must be clearly defined and the settings for data preparation must be reliable.

The first discovered process models should be interpreted carefully by an expert. If they are not satisfying, the steps of data selection and preparation and the application of process discovery algorithms have to be repeated iteratively. Once the process mining algorithm is stable it will be deployed to work automatically.

The material flow can also be used to gain information about the layout of the production system covering therefore a further component of the digital twin (Table. 1): number of machines, conveyors and buffers can be inferred, as well as, to a certain extent, relative positioning of these equipment to each other.

In addition, further process analysis can be performed based on the prepared data. For example, the comparison of time intervals between the activities or throughput times of the entire process often provide deeper understanding of the system behaviour. Additional attributes such as executing resources can be added to the event data for further analysis.

4. Use Case

4.1. Description

This research is part of a research cooperation between the wbk Institute of Production Science at KIT and the central department Connected Manufacturing of the Bosch business unit Powertrain Solutions for diesel, gas and electric drives with the goal to develop an agile production system.

The methodology is validated at a series production of automotive components. The objective is to create a digital twin of a chaku-chaku production line, which consists of an assembly and a testing area. The assembly is done in a classic u-shaped cell. Most of the processes run an automated machine process, but manual labor is still required to insert the product und parts into the machine as well as to remove the part after the machine process. The workers perform some of the transport tasks between machines, other transports are performed by conveyors. Different article types include different parts, which have to be processed at different stations. Therefore, the part routing depends on the variant.

4.2. Implementation

A simulation model of the production line was implemented in Technomatrix Plant Simulation from Siemens.

The machine processing times of the various machines were cleaned and the underlying distribution was obtained with the developed methodology. The assumption that the machine processing times follow a normal distribution was confirmed by a preliminary analysis of different probability distributions on the various machine processing times.

The scrap rate could be calculated directly from the existing test protocols of the testing stations, enabling a direct integration of this information in the simulation model without further data preparation methods.

The deduction of the material flow for different product variants was successfully applied to the industrial application using the event logs of the workstations. These contain the processing steps, duration, result status and its timestamp as well as the part id as process identification. The used production data is stored in a central data base.

The developed Python script directly accesses the data and prepares it automatically. Only faultless process instances are considered. Incomplete and rare process instances are removed by filtering on start and end activities and process length. The process model is discovered by the Alpha Algorithm. For the implementation the Python library PM4Py [24] was used.

Fig. 4 presents a discovered material flow of a product variant. This was derived for three lines of the production system assembly, testing and packaging. The beginning of the assembly shows independent processing steps while the beginning of the testing divides similar processing steps to several workstations.

In addition, routing tables are automatically created from the variant-specific process models for use in the simulation. However, a few production stations do not collect data. These are missing in the discovered process models. For the creation of routing tables, these stations have to be assumed as given.

Further components of the simulation model like scrap rate, failures, and regular activities as for example machine cleaning can also be extracted automatically from the available MES data. The comparison of the digital twin to reality is not yet fully automated, so that the update has to be started manually by the simulation user.

5. Validation

5.1. Behavior and Output

The accuracy of the digital twin to reality was measured by the output over time. The key validation value is the output in a defined time period, for example a shift, a day, or a week. An average difference of output over a week for five simulation runs compared to reality of 0.1% was achieved.

Another important value to consider when validating the model is the spread of the simulation outputs compared to the real spread. Fig. 5 shows that the spread of overall equipment effectiveness (OEE) in the simulation is slightly smaller than the real one, but equally balanced and the mean is very close to the real one. The whisker of the box plot show the last point that is inside interquartile range times 1.5.

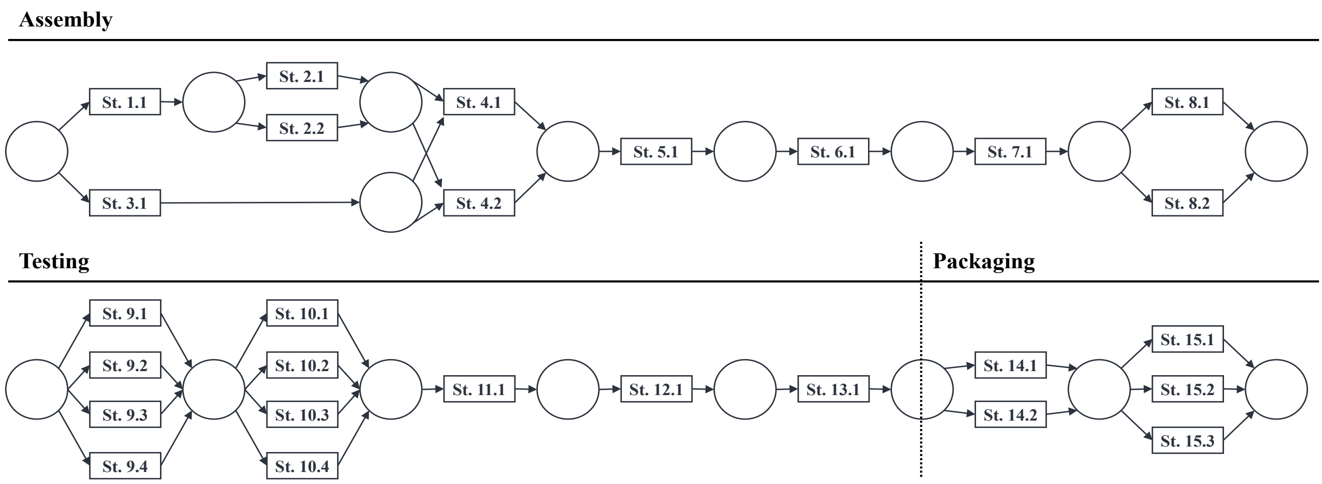

Fig. 4. Discovered material flow for a product variant as petri net

5.2. Transferability

The approach was applied at four lines in the same plant that produce the same product but in different variants. So the system was modelled on a general level based on an exemplary line and then instantiated for each line. This was possible with the chosen approach, even if it is not yet fully automated. Certain steps still have to be done manually, especially data extraction from sources that are difficult to access automatically as for example unstructured data.

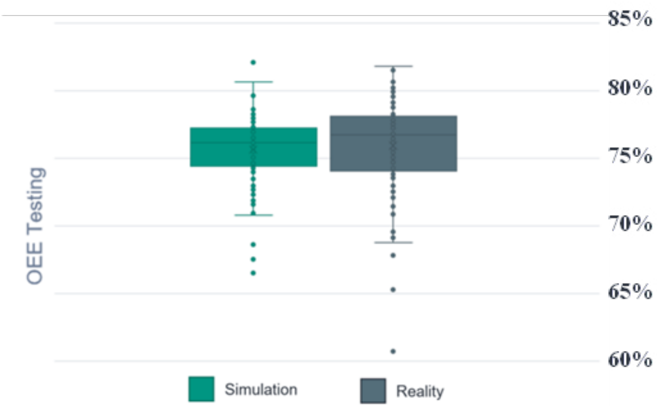

Fig. 5. Boxplots comparing spread of hourly OEE values over one week in simulation and reality (scale was changed)

6. Conclusion and outlook

A methodology to create an adaptive simulation model, which in the end leads to a digital twin of the production system based on regularly updating the model was presented. In particular, the approaches for automated deduction of the material flow by process mining as well as of the machine processing times with the Anderson-Darling-test were explained in greater detail. The fully automated implementation of the presented methodology is not yet completed since some components of the layout and logic cannot yet be identified from the existing data.

Challenges that remain to be solved are the identification of changes in the layout, e.g. new machines and conveyors or their repositioning. To streamline the validation process a method to define the adequate validation time period is important.

The presented approach can be used to ensure a longer usable period of a simulation model which results in higher benefits over time.

Further research topics that arise are the evaluation of the convergence of the simulation model to reality as well of the reaction of the digital twin to changes in the real system. Another possible extension would be the use of the process mining approach for the automated modelling of more components of the dynamic behavior of the production system.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) –project no. 416818066.

References

[1] Stähr T, Englisch L, Lanza G (2018) Creation of configurations for an assembly system with a scalable level of automation. *Procedia CIRP* 76:7–12.
[2] V.D.I. (2014) *Richtlinie 3633, Blatt 1: Simulation von Logistik-, Materialfluß-und Produktionssystemen-Grundlagen*. Beuth, Berlin.
[3] Rosen R, Wichert G von, Lo G, Bettenhausen KD (2015) About The Importance of Autonomy and Digital Twins for the Future of Manufacturing. *IFAC-PapersOnLine* 48(3):567–72.
[4] Barricelli BR, Casiraghi E, Fogli D (2019) A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *IEEE Access* 7:167653–71.
[5] Reese R, Wyatt DL (1987) Software reuse and simulation. in Thesen A, Grant H, David Kelton W, (Eds.). *Proceeding of the 1987 Winter Simulation Conference*, pp. 185–192.
[6] Pidd M (2015 - 2015) Simulation software und model reuse: a polemic. *2015 Winter Simulation Conference (WSC)*. IEEE, pp. 772–775.
[7] Bergmann S (2013) Automatische Generierung adaptiver Modelle zur Simulation von Produktionssystemen, Dissertation, Ilmenau, TU Ilmenau.
[8] Terkaj W, Urgo M (2015) A Virtual Factory Data Model as a Support Tool for the Simulation of Manufacturing Systems. *Procedia CIRP* 28:137–42.
[9] Wenzel S, Weiß M, Collisi-Böhmer S, Pitsch H, Rose O (2008) *Qualitätskriterien für die Simulation in Produktion und Logistik: Planung und Durchführung von Simulationsstudie*. Springer, Berlin, Heidelberg.
[10] Robinson S, Nance RE, Paul RJ, Pidd M, Taylor SJE (2004) Simulation model reuse: definitions, benefits and obstacles. *Simulation Modelling Practice and Theory* 12(7-8):479–94.
[11] Skoogh A, Perera T, Johansson B (2012) Input data management in simulation – Industrial practices and future trends. *Simulation Modelling Practice and Theory* 29:181–92.
[12] Kotiades K (2015 - 2015) Towards self-adaptive discrete event simulation (SADES). *2015 Winter Simulation Conference (WSC)*. IEEE.
[13] Selke C (2005) Entwicklung von Methoden zur automatischen Simulationsmodellgenerierung, Dissertation, München, TU München, iwb.
[14] Bergmann S, Feldkamp N, Strassburger S (2017) Emulation of control strategies through machine learning in manufacturing simulations. *Journal of Simulation* 11(1):38–50.
[15] Yang H, Park M, Cho M, Song M, Kim S (2014 - 2014) A system architecture for manufacturing process analysis based on big data and process mining techniques. *2014 IEEE International Conference on Big Data (Big Data)*. IEEE, pp. 1024–1029.
[16] Rozinat A, Mans RS, Song M, van der Aalst, Wil .M.P. (2009) Discovering simulation models. *Information Systems* 34(3):305–27.
[17] van der Aalst, Wil M.P. (2018) Simulation and Process Mining Match made in Heaven! *Proceedings of the 50th Computer Simulation Conference*.
[18] 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):1016–22.
[19] Negri E, Fumagalli L, Macchi M (2017) A Review of the Roles of Digital Twin in CPS-based Production Systems. *Procedia Manufacturing* 11:939–48.
[20] Brützel O, Overbeck L, Nagel M, Stricker N, Gisela L (2020) Generische Modellierung von halbautomatisierten Produktionssystemen für Ablaufsimulationen. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb* 115(11):792–6.
[21] Stark R, Kind S, Neumeyer S (2017) Innovations in digital modelling for next generation manufacturing system design. *CIRP Annals* 66(1):169–72.
[22] Sun X, Wu C-C, Chen L-R (2018 - 2018) An Automated Warehouse Sorting System for Small Manufacturing Enterprise Applying Discrete Event Simulation. *2018 2nd IEEE Advanced Information Management,Communicates,Electronic and Automation Control Conference (IMCEC)*. IEEE, pp. 1597–1601.
[23] Gutenschwager K, Rabe M, Spiekermann S, Wenzel S (2017) *Simulation In Produktion Und Logistik: Grundlagen und Anwendungen*. Springer Vieweg, Berlin, Heidelberg.
[24] Berti A, van Zelst SJ, van der Aalst, Wil M.P. (2019) Process mining for python (PM4PY) bridging the gap between process-and data science. *arXiv preprint*.