Modular Validation within Digital Twins: A Case Study in Reliability Analysis of Manufacturing Systems

Conference Paper · December 2024

DOI: 10.1109/WSC63780.2024.10838952

2 authors:

Ashkan Zare
University of Southern Denmark

Sanja Lazarova-Molnar
Karlsruhe Institute of Technology

Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

MODULAR VALIDATION WITHIN DIGITAL TWINS: A CASE STUDY IN RELIABILITY ANALYSIS OF MANUFACTURING SYSTEMS

Ashkan Zare[1] and Sanja Lazarova-Molnar[2,1]

[1]Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Odense, DENMARK
[2]Institute AIFB, Karlsruhe Institute of Technology, Karlsruhe, GERMANY

ABSTRACT

As manufacturing rapidly evolves, optimizing processes is essential. Digital Twins, which act as near real-time virtual replicas of the corresponding real-world systems, can support this optimization by providing insights and supporting decision-making. Digital Twins can only be fully effective if their underlying models continuously and accurately reflect the corresponding physical systems. However, not all model components change at the same pace, and relevant data updates also vary in frequency. Thus, Digital Twins require robust validation mechanisms that can identify what parts of models need to be re-extracted, what parts need to be recalibrated, and what parts need to remain same. This is a complex task that necessitates precise partitioning of models with respect to the above noted considerations. Here, we propose a novel approach to modular validation, aimed at supporting Digital Twins. To illustrate our approach, we provide a case study in reliability analysis of manufacturing systems.

1 INTRODUCTION

Simulation has long been used in manufacturing industry as a powerful tool in all stages of product life cycle from design to sales (Lin and Zhou 2018). The continuous evolution of technologies and the ever-changing customer demands has made today's manufacturing processes more complex, and traditional simulation models are limited in their ability to cope with these intricacies. To address the limitations, Digital Twins (DTs), as complex simulation models evolving with their physical counterpart and reflecting the system in near real-time, have been introduced (Grieves 2014). However, to utilize DTs in decision-making, it is essential that they are continuously reflecting the real-life system accurately, and the information derived from them are reliable. For this reason, DT's need to be continuously updated and validated.

Contrary to traditional simulation models, an all-at-once validation approach is not feasible for DT models. As DTs need to reflect the changes to the real system in near real-time, validation becomes an integral part and an enabler of DT development (Zare and Lazarova-Molnar 2024). Therefore, unlike traditional validation techniques, ongoing validation approaches are necessary in ensuring the correct reflection of the system overtime. While studies have been recently published on the subject, a comprehensive approach to validation of DTs is still missing in literature.

continuously validating

We use a case study from reliability modeling of manufacturing systems to motivate and illustrate our approach to validation. A complex manufacturing system consists of different machines and production processes, and validating the corresponding reliability model as a single unit may not be optimal. In case of discrepancies, re-extracting the model completely not only would be complex and resource exhaustive, but depending on the validation policies and consequent recalibration, it may also lead to losing parts of the original model. Moreover, if only the final output as opposed to the collection of individual processes leading to the output is considered, internal changes to the processes may not be detected. Figure 1 illustrates these challenges. Let us assume a sequential production line consisting of two main interchangeable operations, both susceptible to faults leading to failures. If at a certain point in time the production line is reconfigured and the order of the operations is changed without affecting the final output, a validation process should still be able to detect the change in the underlying processes in the model. Furthermore, since only the order of the operations has changed, each operations' process and its fail/repair time are intact and should be kept. Therefore, the validation process must also detect the parts of the model that need to be preserved, recalibrated, or re-extracted, instead of re-extracting the whole model.

Modellbestandteile konstant setzen die bei der Modularen Veränderung nicht geändert werden

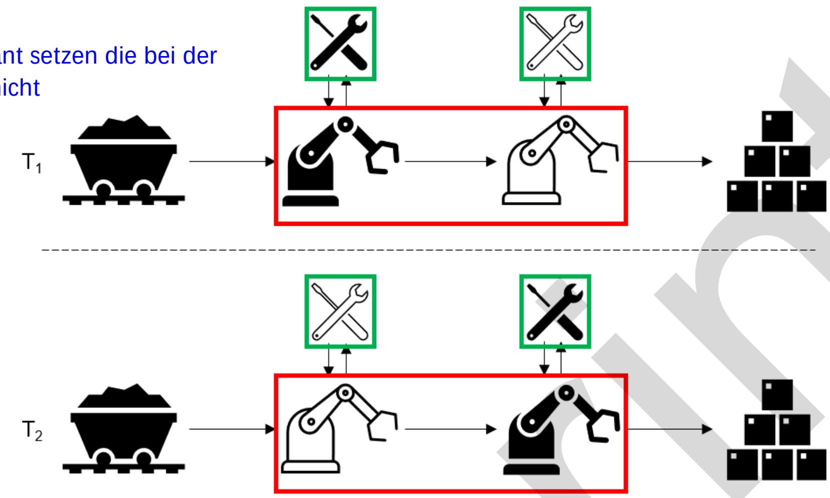

Figure 1: A sequential production line (T1) and its reconfigured set up (T2).

Reliability of a system is defined as the probability of a system performing its intended function without failure over a period of time (Bazovsky 2004). Analyzing reliability of a manufacturing system is crucial as it helps identify the critical components, processes, failures, and the overall system performance. In reliability modeling and simulation, it is crucial to consider the significantly varying frequencies of events in the validation process of a reliability model. E.g., as the gap between rate of occurrence of failures and the rate of occurrence of production processes is large, fault models need a longer timeframe to be extracted compared to production process models. Moreover, in the dynamic world of today's manufacturing, production processes go through changes more often, and their models need to be validated and updated more regularly in comparison with fault models where the models are more static. Depending on these diverse frequencies, certain parts may require preservation while others may need regular recalibration or re-extraction.

To address the challenges of detecting changes in the underlying models as well as when parts of a model need preservation, recalibration, or re-extraction, complex systems and their digital counterparts need to be treated as collections of modules to facilitate the individual and collective validation processes. We propose a modular approach to validation and model recalibration/re-extraction, building on the two-phase validation framework proposed by Friederich and Lazarova-Molnar (2023). In our expanded two-phase validation approach, the model is first validated based on historical data, and in the second phase, the model is divided into modules and the continuous validation process preserves, recalibrates, and re-extracts each module based on their specific characteristics.

The paper is organized as follows. In Section 2, we review the background and the current approaches to validation of DTs, model partitioning, and reliability of systems. Section 3 focuses on our modular approach to validation. In Section 4, we further demonstrate our approach through a case study in reliability modeling. Finally in Section 5, we conclude our findings and identify challenges and future directions.

Erst: Validierung durch historische Daten, dann werden in einer kontinuierlichen Validierung immer neue Modellbestandteile gebildet die dann entweder abgekapselt werden oder verändert werden und neu gebildet werden.

2 BACKGROUND

In the following, we review the current literature on validation of DTs and highlight the need and the existing gap in modular approaches, provide an overview of model partitioning, and finally describe reliability modeling and analysis.

2.1 Validation of Digital Twins

Ensuring DT models accurately reflect their real-world counterparts over time is vital for the core functionality and purpose of DTs. Validation is the process of confirming a model's output, within its intended purpose, closely matches the real system's performance. Although validation of simulation models is not a new topic of research, most of the existing approaches, as described by Sargent (2013), are applicable to traditional and static simulation models. DT models' objective of reflecting corresponding systems in near real-time seeks a continuous approach and conformance analysis of the model (Hua et al. 2022).

Recent studies, albeit with a limited coverage and applicability, have explored approaches for continuous validation within DTs. An approach to DT validation based on combining human expert knowledge and data collected from IoT devices was proposed by Hua et al. (2022). Similarly, Lugaresi et al. (2023) proposed a continuous validation process by comparing the output of the model to sequence of data gathered from the system and obtaining a similarity level. Using approximation function for continuous validation of DT models was another approach presented by Mertens and Denil (2023). Using machine learning and control charts, specifically K-nearest neighbor and p-control chart, dos Santos et al. (2023) proposed a periodic validation approach. Overbeck et al. (2023) introduced a periodic validation process based on automatic comparison of key performance indicators (KPIs) from the real system and its virtual counterpart. Continuous calibration of DTs by particle filtering is another approach proposed by Ward et al. (2021). Friederich and Lazarova-Molnar (2023) proposed a two-phase validation approach, illustrated in Figure 2, in which the first phase is the initial validation of the model based on historical data and standard validation techniques, and the second phase is the continuous validation based on streaming data and comparison of the real system's output and the model's output.

All reviewed approaches aim at validating DT models as a whole. These approaches do not consider the need for a modular validation process, preserving and calibrating each module depending on its requirements. As noted in Section 1, we incorporate our proposed modular validation approach in the two-phase validation framework proposed by Friederich and Lazarova-Molnar (2023).

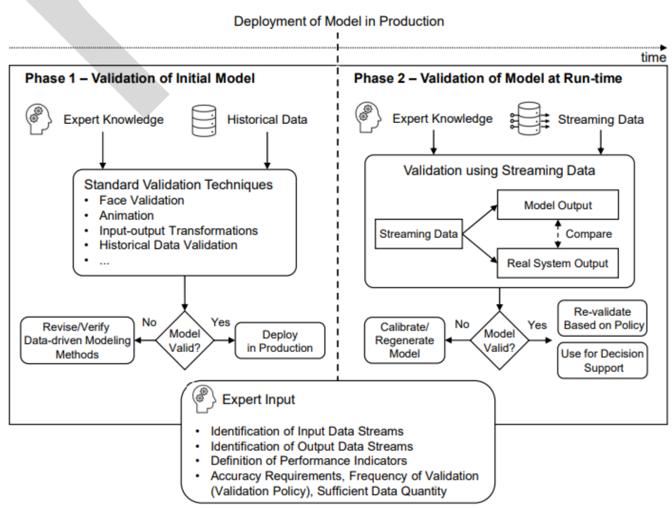

Figure 2: Two-phase validation approach proposed by Friederich and Lazarova-Molnar (2023).

2.2 Partitioning Simulation Models

Simulation models are powerful tools, helping in exploring and understanding real-life systems. They facilitate decision-making, performance optimization, and risk mitigation. However, as systems grow increasingly intricate, simulation models replicating their behaviors are also evolving and becoming more complex. Running these complex simulations at once is computationally resource exhaustive and time consuming. Model partitioning aims at alleviating this difficulty by dividing a complex model into multiple components. We investigated model partitioning for research on meaningful ways to partition models that can be utilized for modular validation.

Studies on model partitioning have been published, however, with a focus on time and computational optimization. In that sense, GloMoSim library (Zeng et al. 1998) decomposes simulation models into smaller components to evenly distribute computational loads. Similarly, Myers et al. (2016) proposed an approach to reduce data transfer and storage requirements during simulation by utilizing an online in situ method in identifying important time steps and variables. An adaptive solution based on self-clustering of autonomous agents and re-allocation of the partitioned simulation components (agents) aiming to reduce communication and computational load was proposed by D'Angelo (2017). Another approach aiming at simulation efficiency was proposed by Papadopoulos and Leva (2015), utilizing principles of dynamic decoupling to partition a model based on relevant time scales.

Partitioning simulation models helps reduce computational resource consumption while running the complex models. However, our study diverges from this focus on consumption optimization. Our aim for partitioning is to preserve, and re-use certain parts of the model during its continuous updating as part of the DT setup. In addition, as proposed by Netter et al. (2013), model partitioning can be utilized in validation of the system through intermediate, sub-system validation.

2.3 Reliability Modeling and Analysis

In the following, we provide an introduction to reliability modeling and analysis. Reliability analysis is a highly relevant use case for the need of modular validation, especially because faults are rare events, so gathering relevant data is not a trivial task (Parhizkar and Mosleh 2022; Niloofar and Lazarova-Molnar 2021). Bazovsky (2004) defines reliability of a system as "probability of a device performing its purpose adequately for the period of time intended under the operating conditions encountered". Reliability assessment is the evaluation of performance of a system to ensure its operation without failure (Friederich and Lazarova-Molnar 2024). Reliability of a system is generally assessed in three main categories: hardware reliability, software reliability, and human reliability (Hoyland and Rausand 2009). Reliability modeling enables understanding and estimating reliability of a system resulting in improved production management and maintenance strategy (Li et al. 2021). Here, our focus is on hardware reliability, namely machinery and equipment.

Manufacturing systems are comprised of machines and equipment, and with any hardware equipment failure is imminent. Failure in the components can lead to downtime in the production process, and even accidents. Reliability, maintainability, and availability of a system relies on its hardware which makes hardware reliability assessment essential. IEEE defines the three attributes as a system's ability to perform its function without failure over time, the efficiency with which a system can be repaired after a failure, and the proportion of time a system is operational (Geraci 1991).

Many articles have been published on hardware reliability assessment of manufacturing systems utilizing various modeling formalisms. Mubarok et al. (2018) proposed a hierarchical assessment evaluating reliability of the system at four different levels, component, machine, system, and cloud level. A method to determine critical equipment based on occurrence of failure was proposed by Relkar (2021). Adamyan and He (2002) proposed a method utilizing Petri net modeling in assessing reliability and safety of manufacturing system with sequential failures.

In complex manufacturing systems of today, reliability analysis is essential in ensuring systems' performance. Moreover, reliability models include models of parts such as failures that are relatively fixed and not frequently changing which highlights the need for modular validation and preserving of certain parts.

3 MODULAR VALIDATION

In this section, we present our modular approach for validation of DTs. Figure 3 illustrates our proposed approach for modular validation. Similar to the original two-phase validation approach (Friederich and Lazarova-Molnar 2023), the initial extracted model is validated using standard validation techniques such as historical data validation and face validation, in the first phase. In case of discrepancies, further examination and revision needs to be performed to ensure the extracted model accurately reflects the real system process. The second phase is the continuous modular validation process at run-time where the initial model is divided into sub-models, and each sub-model is validated separately based on its validation policy. In the following subsections, we elaborate on the partitioning, validation policy, and the recalibration process.

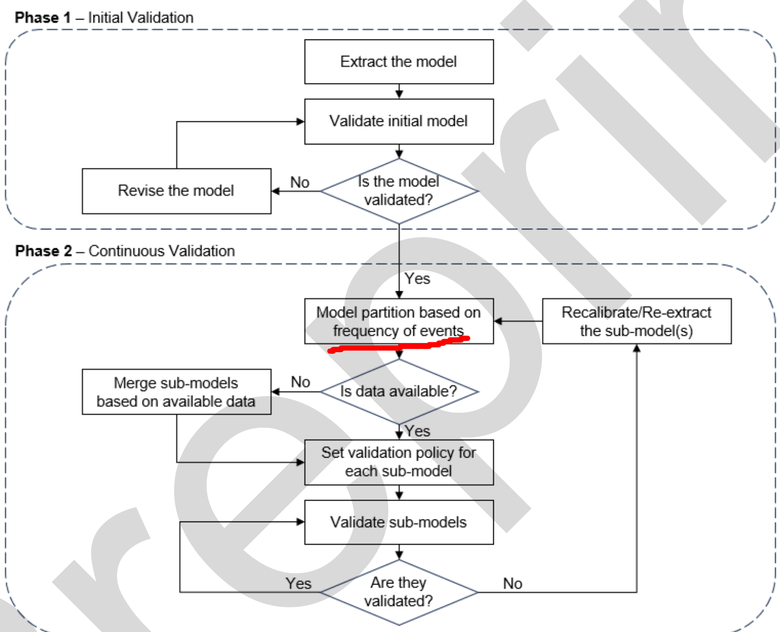

Figure 3: Modular validation process.

3.1 Model Partitioning for Validation

To ensure a robust continuous validation process, the initially validated model is partitioned into smaller sub-models. Breaking down the model allows for targeted validation of the underlying sub-models as well as facilitates validation of complex models. In our approach, we use stochastic Petri nets as the modeling formalism. A stochastic Petri net can be extracted from event logs of the system containing timing information (Rogge-Solti et al. 2014). In addition, focusing on event logs is beneficial as interactions and dependencies among different processes are captured in the event logs and ultimately reflected in the stochastic Petri net model. This is essential to partitioning as it ensures that the sub-models, though validated individually, are not treated as complete standalone models. Without consideration of the interactions, every sub-model can be valid while the complete model remains invalid. As noted in the previous section, the criterion for partitioning simulation models is mostly computational cost. However, our criteria for partitioning are the significance of events and their rate of occurrence, as well as comparable data availability. In a Petri net, transitions are used to describe events that occur in the system, and we look at the frequency of the firing of the transitions to calculate the rate of occurrence and a transition's significance. Moreover, as event logs may be missing events represented by corresponding transitions, we partition the model in a way that equivalent data from the system is available to evaluate validity of the sub-models.

The partitioning process is a bottom-up process in which the model is first divided based on the transitions. For each individual transition, its rate of occurrence is calculated to determine if it is a frequent activity. Depending on the frequency of occurrence, we prioritize if the transition and its associated arcs and places need to be preserved or if it is necessary to regularly investigate for modifications. To confirm transitions correctly represent the real events, corresponding data from the real system must be available. However, real system limitations may hinder data collection and availability. To mitigate this challenge, inspired by (Netter et al. 2013), for every transition in the model, we look for corresponding event data from the real system. In case of availability of data, the transition is individually assessed for validation. While in the case of missing data, the transition is combined with its preceding and succeeding transition creating a composite sub-model which can be compared to corresponding data from the real system.

3.2 Frequency of Validation

Depending on the objective of the model, the model's accuracy demands may differ. For example, a model focusing on safety requires a high level of accuracy, and it needs a more precise validation process. Moreover, sufficient data must be available to validate a model as validation based on insufficient data may result in inaccurate models and an incorrect reflection of the system. Frequency of validation is an important step in defining the requirements of a continuous validation process from quantity of data needed for a reliable validation process to the rate at which validation process is conducted.

Rate of occurrence of events is a measure by which frequency of validation can be determined. Frequent processes, such as production processes, may require more regular validation to ensure an accurate reflection of the production. While infrequent events, such as failures, require less validation as they are rare events and the sufficient data needed for validation may take time to be collected. Therefore, for every individual or composed sub-model, a validation policy is set based on its occurrence frequency stating how often validation process needs to be performed. This means that at every iteration, the DT model's accuracy is investigated through validation of a subset of the underlying sub-models. This modular approach not only ensures a precise representation of the system, but also facilitates a more cost and time efficient validation process.

3.3 Model Recalibration and Re-extraction

Recalibration/re-extraction is another essential step in continuous validation processes of DTs as a system may change throughout its lifecycle. This iterative process addresses these changes and reduces the discrepancies between the model and the real system ensuring the model closely reflects the system's functioning over time. However, determining what simulation models' parameters need to be adjusted as well as deciding on when recalibration is no longer feasible, and model re-extraction is needed, is challenging. The modular approach aids in the recalibration process as only sub-models where discrepancies are detected need to be considered. Therefore, the parameters to be adjusted are confined within the sub-models where discrepancies are detected. Moreover, the model re-extraction process is made easier as the re-extraction applies to the sub-model, or a subset of sub-models affected by the discrepancies.

Defining what constitutes a discrepancy is contingent upon the goals of the model and the requirements and limitations of the users. For instance, in a model prioritizing safety, the margin of error must be small as even a slight deviation from the real system could result in accidents and safety risks. In contrast, a model prioritizing production throughput may be more lenient towards minor variations. In our proposed approach, two thresholds are established for the recalibration/re-extraction process. The first threshold is used to identify if a discrepancy has occurred by comparing the outcome of the sub-model with that of the actual system. If the result falls outside the threshold limits, the recalibration process is initiated. The second threshold limits the number of recalibration iterations a sub-model can undergo. If the discrepancy remains and recalibration process fails to yield satisfactory results, it may indicate that recalibration approach is not effective, and re-extraction is required. Algorithm 1 presents a high-level narrative form algorithm for our proposed approach.

Algorithm 1 Algorithm for Modular Validation.
Input: stochastic Petri net model, event log from the real system, validation policy, discrepancy threshold, recalibration threshold.
Output: recalibrated/re-extracted sub-models.

Step 1 Calculate frequency of occurrence of transitions (events).

Step 2 Flag each transition for preservation or regular validity checking based on its occurrence frequency.

Step 3 Partition the model based on the preservation/regular validity checking flag into set of sub-models.

Step 4 For every transition, find corresponding data from the event log. If no corresponding event data is found, combine the transition with its preceding and/or succeeding transitions creating a grouped sub-model.

Step 5 Based on the validation policy and the discrepancy threshold, validate the sub-model.

Step 6 If the results are outside the discrepancy threshold boundaries, start the recalibration process.

Step 7 If the results are still not within the discrepancy threshold boundaries, go to **Step 6** while the recalibration threshold is not reached.

Step 8 If the recalibration threshold is reached and no acceptable result is found, re-extract the sub-model.

4 CASE STUDY

To demonstrate our approach for modular validation of DTs, we conducted two experiments using reliability models. As noted earlier, reliability models are highly adequate examples because of the significant varying frequencies of occurrence of events.

Our first case study is a continuous validation process of a simple sequential manufacturing system where without preservation, the recalibrated model may fail to correctly represent the real system. In the second case study, we focus on the need for intermediate validation points as opposed to validation of final output.

4.1 Modularity for Preservation of Model Parts

In the approach proposed by Friederich and Lazarova-Molnar (2023), the whole model is re-extracted when a discrepancy is detected. However, depending on the frequency of the validation process, key events may not occur during the validation timeframe, and the corresponding re-extracted model may not correctly reflect the system.

In this experiment, we considered a simple wood manufacturing system consisting of two interchangeable and sequential operations, drilling and sanding, and an AGV responsible for transferring the material. To develop the initial reliability model, we applied the model extraction approach based on event data and process mining presented by Friederich and Lazarova-Molnar (2022). The initial extracted model based on a month-long event data, illustrated in Figure 4 as stochastic Petri net, was then validated by comparing its production throughput (number of completed products) to that of the system.

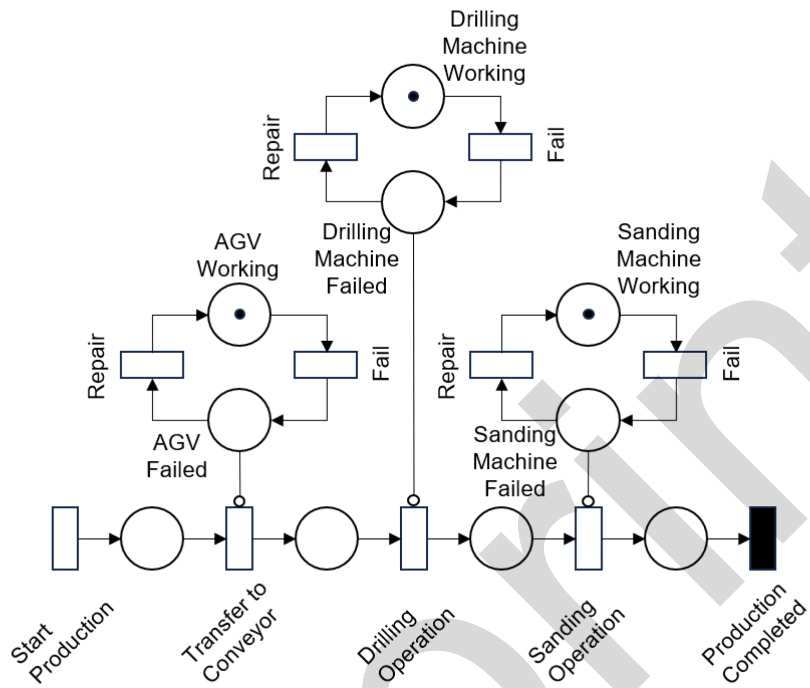

Figure 4: Initial extracted reliability model of the system.

After the extraction and validation of the initial model, we calculated the frequency of occurrence of events. As expected, activities related to production processes, such as transfer of material and operations, were more frequent than failures and repairs. Therefore, the model was divided to sub-models of production processes and failures/repairs.

To continuously validate our model, we set a daily validation policy for the production throughput. Daily events of the real system were captured and the system's productivity KPI was calculated. To generate the corresponding result from the model, we conducted 100 terminating simulation replications based on the validation policy. The discrepancy threshold was set based on the confidence interval generated from the simulation runs. If the result from the system was outside the confidence interval boundaries, the recalibration process was initiated only for frequent events, namely the production sub-models, preserving failures/repairs. Moreover, the limit for recalibration attempts was set to five before the sub-model was re-extracted from the daily logs.

To better highlight the need for model parts preservation, we also conducted a daily re-extraction of the model. Depending on the events that occurred during the day, the daily extracted models were inconsistent. For example, on day 3, when the system had no failures, the extracted model only captured the production processes, completely losing the fault models. Comparing the result of daily extraction with our proposed approach, presented in Figure 5, we found that our approach and the recalibration of the sub-models keeps the model consistently accurate while daily re-extraction of the model depending on the data available on the day yields inconsistent models, losing both parts of the model, that may have been unavailable on the day, and accuracy.

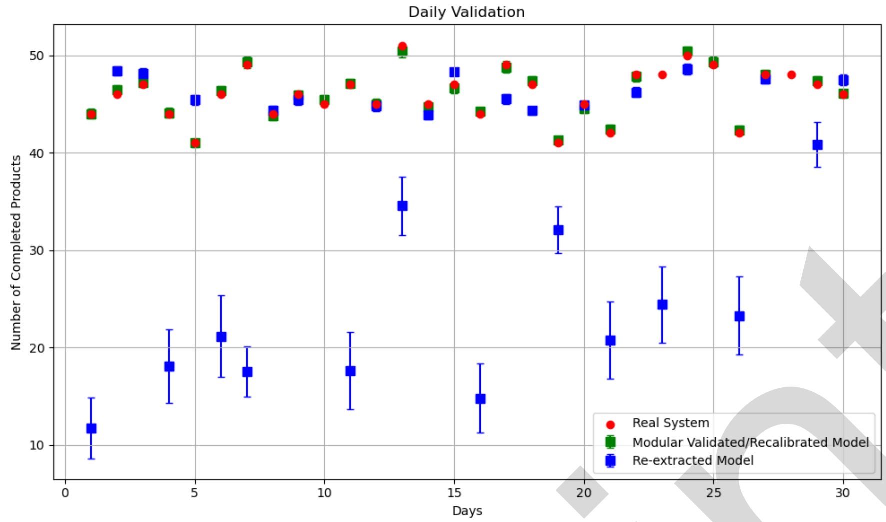

Figure 5: Results of daily modular validation and daily model re-extraction (error bars representing 95% confidence interval for 100 independent replications).

4.2 Modularity for Intermediate Validation

In today's dynamic manufacturing, the production lines might frequently undergo modifications, and DTs should be able to reflect these changes. Validating the underlying components of DTs, and intermediate validation points aids in achieving this objective. To demonstrate the need for intermediate validation, we used the same manufacturing system from the previous experiment. Similarly, we extracted and validated the initial model, calculated the frequency of occurrence, and divided the model to sub-models of production processes and failures/repairs. However, we alternated the order of the two production operations of the system and captured the new event data. As mentioned earlier, the two production operations are interchangeable, and the order of their operations does not affect the final output of the production.

Focusing only on the final throughput of the model and the altered system, the original model was still deemed valid. However, when comparing the sub-models, a discrepancy in the production processes was detected. Sanding is a longer process than drilling, and when altered, throughput of drilling is bottlenecked by the sanding process. This bottleneck and the interaction between the two operations is captured in the event log, making it possible to detect the discrepancy. Figure 6 illustrates the validation result of the model highlighting the discrepancy in the drilling operation.

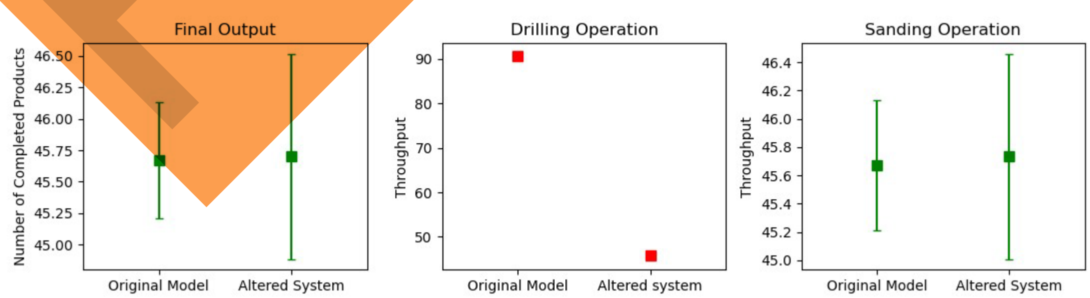

Figure 6: Results of validation of sub-models (production processes) (error bars representing 95% confidence interval for 100 independent replications).

To reflect the change of the real system in the model, recalibration was initiated with the recalibration threshold of five attempts. The recalibration process was unable to achieve results within the accepted discrepancy threshold, and therefore the sub-model was re-extracted using the altered system's logs while the unchanged processes, such as fault models, were preserved. Finally, the new estimated distribution times of events in the model, derived from the altered system's logs, were set to the model, and the model's performance was compared to the actual system, illustrated in Figure 7.

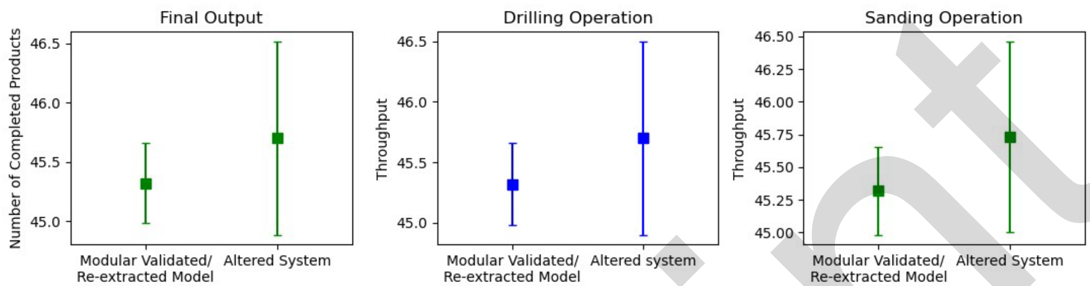

Figure 7: Results of validation of sub-models (production processes) after sub-model re-extraction (error bars representing 95% confidence interval for 100 independent replications).

As evident by the results, using a modular approach to validation can aid in detection of discrepancies in the underlying models of a complex DT model. Moreover, we showed that considering the interaction among the sub-models derived from the event log is critical in ensuring the complete model validity. Finally, the model was validated and accurately reflected the system without the need of complete re-extraction as only the parts, where discrepancy was apparent, were re-extracted and the consistent parts were left intact.

5 CONCLUSION

Digital Twins are a valuable tool in the manufacturing industry as they enable optimization and decision-making support. Utilizing the full potential of Digital Twins depends heavily on having a reliable and valid model of the real system. However, validation of these models is a challenging task, especially in manufacturing systems of today as their complex and dynamic nature requires a robust and continuous model validation approach capable of distinguishing the significance and characteristics of the underlying models to accurately reflect the real-system's intricacies over time. Depending on the underlying model, a validation process must be able to detect if it or some of its parts need to be preserved, recalibrated, or re-extracted to correctly represent its real-world counterpart.

In this paper, we proposed a modular approach to the validation problem, building on the two-phase validation framework by Friederich and Lazarova-Molnar (2023). Similar to their approach, the first phase is the initial validation of the model based on standard validation techniques. In the second phase, however, we propose a partitioning algorithm based on frequency of occurrence of events and data availability to divide the model into sub-models. Furthermore, we focus on the events captured in the event log, making detection of interaction among the sub-models possible resulting in validation of the complete model. Every sub-model, depending on its rate of occurrence, is given a validation policy by which it is continuously validated over time. In case of detecting discrepancy in the validation process, the sub-model under question goes through a recalibration process and if the recalibration fails to achieve satisfactory results within the recalibration attempt threshold, the sub-model is re-extracted. Finally, through an example of reliability modeling, we examined our approach and further demonstrated the need for preservation of parts and intermediate validation of the model.

While we have addressed several challenges in continuous validation of Digital Twins and the need for modularity, it is important to recognize certain challenges and limitations remain to be addressed to achieve a robust validation mechanism. For the partitioning of the model and the validation policy, we only used the data available and focused on the frequency of occurrence of events. However, incorporating expert knowledge in the partitioning and validation policy processes may yield more optimized sub-models and ultimately an accurate model. In the recalibration process, we used a simple threshold to determine when re-extraction is necessary. Deciding on when to recalibrate or re-extract is a complex challenge that may need a more comprehensive approach. Moreover, we implemented an iterative optimization method for the recalibration process. However, the challenge of selecting an appropriate recalibration algorithm capable of achieving satisfactory result persists. As future work, we suggest further testing of the proposed approach on more complex systems, and against other applicable validation approaches, as well as integration of expert knowledge in the process.

ACKNOWLEDGMENTS

This research is part of the ONE4ALL project, funded by the European Commission, Horizon Europe Programme under the Grant Agreement No 101091877.

REFERENCES

Adamyan, Angela, and David He. 2002. 'Analysis of sequential failures for assessment of reliability and safety of manufacturing systems', *Reliability Engineering & System Safety*, 76: 227-36.

Bazovsky, Igor. 2004. *Reliability theory and practice* (Courier Corporation).

D'Angelo, Gabriele. 2017. 'The simulation model partitioning problem: an adaptive solution based on self-clustering', *Simulation Modelling Practice and Theory*, 70: 1-20.

dos Santos, Carlos Henrique, Afonso Teberga Campos, José Arnaldo Barra Montevechi, Rafael de Carvalho Miranda, and Antonio Fernando Branco Costa. 2023. 'Digital Twin simulation models: a validation method based on machine learning and control charts', *International Journal of Production Research*: 1-17.

Friederich, Jonas, and Sanja Lazarova-Molnar. 2022. "Data-driven reliability modeling of smart manufacturing systems using process mining." In *2022 Winter Simulation Conference (WSC)*, 2534-45. IEEE.

———. 2023. "A Framework for Validating Data-Driven Discrete-Event Simulation Models of Cyber-Physical Production Systems." In *2023 Winter Simulation Conference (WSC)*.

———. 2024. 'Reliability assessment of manufacturing systems: A comprehensive overview, challenges and opportunities', *Journal of Manufacturing Systems*, 72: 38-58.

Geraci, Anne. 1991. *IEEE standard computer dictionary: Compilation of IEEE standard computer glossaries* (IEEE Press).

Grieves, Michael. 2014. 'Digital twin: manufacturing excellence through virtual factory replication', *White paper*, 1: 1-7.

Hoyland, Arnljot, and Marvin Rausand. 2009. *System reliability theory: models and statistical methods* (John Wiley & Sons).

Hua, Edward Y, Sanja Lazarova-Molnar, and Deena P Francis. 2022. "Validation of Digital Twins: Challenges and Opportunities." In *2022 Winter Simulation Conference (WSC)*, 2900-11. IEEE.

Li, Kewu, Jun Yang, and Shuo Huang. 2021. 'A Review on Reliability Modeling of Manufacturing Systems Considering the Production Quality and Equipment Reliability', *Recent Patents on Engineering*, 15: 22-29.

Lin, Zhang, and Longfei Zhou. 2018. 'Modeling & simulation technology in manufacturing', *Journal of System Simulation*, 30: 1997-2012.

Lugaresi, Giovanni, Sofia Gangemi, Giulia Gazzoni, and Andrea Matta. 2023. 'Online validation of digital twins for manufacturing systems', *Computers in Industry*, 150: 103942.

Mertens, Joost, and Joachim Denil. 2023. "Digital-twin Co-evolution using Continuous Validation." In *Proceedings of the ACM/IEEE 14th International Conference on Cyber-Physical Systems (with CPS-IoT Week 2023)*, 266-67.

Mubarok, Khamdi, Xun Xu, Xinfeng Ye, Ray Y Zhong, and Yuqian Lu. 2018. 'Manufacturing service reliability assessment in cloud manufacturing', *Procedia CIRP*, 72: 940-46.

Myers, Kary, Earl Lawrence, Michael Fugate, Claire McKay Bowen, Lawrence Ticknor, Jon Woodring, Joanne Wendelberger, and Jim Ahrens. 2016. 'Partitioning a large simulation as it runs', *Technometrics*, 58: 329-40.

Netter, Florian, Frank Gauterin, and Björn Butterer. 2013. "Real-data validation of simulation models in a function-based modular framework." In *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*, 41-47. IEEE.

Niloofar, Parisa, and Sanja Lazarova-Molnar. 2021. "Data-driven modelling of repairable fault trees from time series data with missing information." In *2021 Winter Simulation Conference (WSC)*, 1-12. IEEE.

Overbeck, Leonard, Stephen C Graves, and Gisela Lanza. 2023. 'Development and analysis of digital twins of production systems', *International Journal of Production Research*: 1-15.

Papadopoulos, Alessandro Vittorio, and Alberto Leva. 2015. 'A model partitioning method based on dynamic decoupling for the efficient simulation of multibody systems', *Multibody System Dynamics*, 34: 163-90.

Parhizkar, Tarannom, and Ali Mosleh. 2022. "Guided probabilistic simulation of complex systems toward rare and extreme events." In *2022 Annual Reliability and Maintainability Symposium (RAMS)*, 1-7. IEEE.

Relkar, Anand S. 2021. 'Risk analysis of equipment failure through failure mode and effect analysis and fault tree analysis', *Journal of Failure Analysis and Prevention*, 21: 793-805.

Rogge-Solti, Andreas, Wil MP van der Aalst, and Mathias Weske. 2014. "Discovering stochastic petri nets with arbitrary delay distributions from event logs." In *Business Process Management Workshops: BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*, 15-27. Springer.

Sargent, Robert G. 2013. 'Verification and validation of simulation models', *Journal of simulation*, 7: 12-24.

Ward, Rebecca, Ruchi Choudhary, Alastair Gregory, Melanie Jans-Singh, and Mark Girolami. 2021. 'Continuous calibration of a digital twin: Comparison of particle filter and Bayesian calibration approaches', *Data-Centric Engineering*, 2: e15.

Zare, Ashkan, and Sanja Lazarova-Molnar. 2024. "Validation of Digital Twins in Labor-Intensive Manufacturing: Significance and Challenges." In *The 7th International Conference on Emerging Data and Industry (EDI40)*. Hasselt, Belgium.

Zeng, Xiang, Rajive Bagrodia, and Mario Gerla. 1998. "GloMoSim: a library for parallel simulation of large-scale wireless networks." In *Proceedings of the twelfth workshop on Parallel and distributed simulation*, 154-61.

AUTHOR BIOGRAPHIES

ASHKAN ZARE is a PhD student at the Mærsk Mc-Kinney Møller Institute, University of Southern Denmark. His research interests concern Modeling and Simulation, Data Analytics, and Machine Learning. His PhD project focuses on continuous validation of Digital Twins in manufacturing systems. His email address is zare@mmmi.sdu.dk.

SANJA LAZAROVA-MOLNAR holds two full professorships, at the Karlsruhe Institute of Technology, and the University of Southern Denmark. Her research focuses on data-driven simulation, digital twins, and cyber-physical systems modeling for reliability and energy efficiency enhancement. Actively engaged in developing advanced methodologies, she leverages her expertise to optimize complex systems through data-driven simulation. She leads activities focused on digital twins and data-driven simulation in several European and national projects. Furthermore, Professor Lazarova-Molnar assumes leadership roles in IEEE and The Society for Modeling & Simulation International (SCS), contributing significantly to these professional organizations. She was also one of the Proceedings Editors for the Winter Simulation Conference in 2019 and 2020 and an associate editor of SIMULATION: Transactions of The Society for Modeling and Simulation International. Her email address is sanja.lazarova-molnar@kit.edu.