Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 136 (2025) 61–66

www.elsevier.com/locate/procedia

35th CIRP Design Conference

# Application of artificial intelligence in model-based systems engineering of automated production systems

Timo Schuchter[a], Patrick Saft [a], Ralf Stetter[a,*], Markus Pfeil[a], Wolfram Höpken[a], Markus Till[a], Stephan Rudolph[b]

[a] Ravensburg-Weingarten University (RWU), Doggenriedstraße, 88250 Weingarten, Germany
[b] Institute of Aircraft Design, University of Stuttgart, Pfaffenwaldring 31, 70569 Stuttgart, Germany

* Corresponding author. Tel.: 00497515019822. E-mail address: ralf.stetter@rwu.de

**Abstract**

Despite the incontestable appeal, the application of artificial intelligence (AI) in engineering processes is still limited to isolated applications and, in some fields, enthusiasm has given way to disillusionment. This paper aims to contribute to a concept of a framework that allows the application of AI in model-based systems engineering (MBSE) processes of automated production systems; the main focus is hereby on the MBSE processes. The aim of the complete framework is to realize an AI-based, self-learning digital twin that automatically adapts to the real system behavior and represents an optimal image of a product and its production process at all times. An expressive, semantic overall model serves as the basis for new approaches to artificial intelligence. In the complete framework, knowledge gained using AI methods is integrated into the overall model and thus brought into an overall context. Such an overall model improves the interpretability and explainability of the AI models and enables complex analyses, simulations and forecasts. The core element of the approach is a novel, AI-based, self-learning engineering model consisting of a product and production model that maps function, behavior and product geometry. Graph-based design languages are used for forming a central data model and functional mock-up units are applied for continuous co-simulation. The approach is underlined by means of an application to the design of automated assembly systems.



*Keywords:* artifical intelligence, model based systems engineering; behavior modelling; design methodology, tools and technologies

## 1. Introduction

Production companies face constantly numerous challenges because of, amongst others, increasing customer expectations, global competition and raising sustainability efforts. Several industries undergo currently fundamental changes, for instance the automotive industry, which is realizing the transformation towards electro-mobility and autonomous driving. Artificial intelligence (AI) is considered a high-potential approach to support companies in this challenge; AI may support several processes in production companies and can have a significant impact on the digital transformation of organizations [1].

However, whereas the media coverage on the use of AI in recent years was very positive, it is indicated that in some sectors enthusiasm has given way to disillusionment [2]. It is important to note that AI implementation requires sophisticated processes, changes in established practices and management styles [1]. The research described in this paper intends to propose a concept of a framework which allows the application of AI in and for model-based systems engineering processes of products and the respective automated production systems. This framework is based on a novel, AI-based, self-learning engineering model which integrates product and production models which map function, behavior and product geometry.

2212-8271 © 2025 The Authors. Published by Elsevier B.V.

10.1016/j.procir.2025.08.013

In this framework, graph-based design languages (GBDLs) are applied in order to achieve a consistent and semantically rich central data model and functional mock-up units (FMUs) are applied in order to enable continuous co-simulation. The GBDLs can be applied to include information in form of ontologies, to enable the storage in a central data model and to realize an integrated development environment by employing the software Design Compiler 43™ (DC43) [3]. It is intended to apply functional mock-up units (FMUs) [4] in order to enable a flexible, modularized co-simulation framework for the integrated development process.

In order to be able to address these intentions, the following research question needs be answered:

*How can an engineering framework based on model-based systems engineering be designed to enable an effective and efficient application of artificial intelligence in all stages of an integrated product and automated production system development process?*

The different stages of an integrated product and productions system processes are visible in Figure 1.

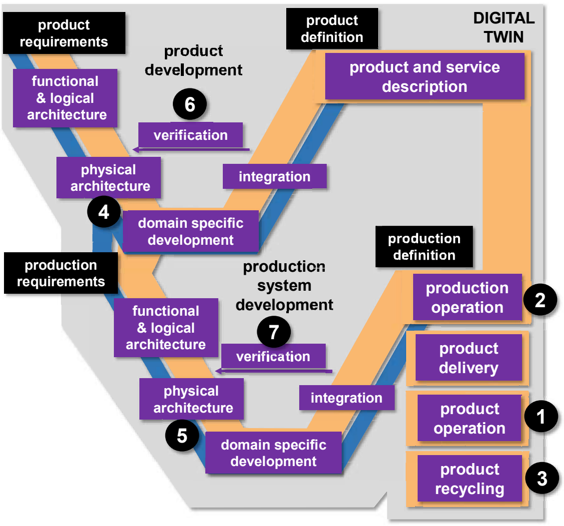


Fig. 1. Process depiction of an integrated product and production development process

The main element of the process shown in Fig. 1 are two connected V-models as suggested e. g. by VDI/VDE2206 [5]. The first V-model describes the typical development process of a mechatronic or cyber-physical product with the determination of the functional, logical architecture and physical architecture, the domain specific development of modules and the integration and verification. Similarly, the second connected V-model describes the analog activities for the production system.

The interconnected V-models lead to a product and service description, to the actual production operation, the product delivery and its operation as well as for the recycling of the product. Therefore, the most important areas of activity in the development and operation of a product and its respective production system are covered. Additionally, in Fig. 1 the numbers 1 to 7 indicate areas in the integrated product and production development process within which an application of AI may be sensible. These possible applications are explained in the following list.

1. One main field for the application of AI is during the operation of the product. Several research initiatives explore the application of AI in the operation of autonomous and semi-autonomous cars in areas such as advanced driver assisting systems, car emissions, predictive maintenance, security, connected vehicles, and driver monitoring systems [6]. In electronic products such as smartphones, AI is already used in several practical applications such as image editing.
2. The second most obvious field for the application of AI is the operation of production systems. Possible use cases are the detection of faulty assembly processes e. g. by means of defect-detection strategies [7], [8], preventive maintenance through anomaly detection and energy efficiency optimization [9]. A current study was able to identify 20 different use cases for AI in the car production process [2]. Several research and implementation groups are currently addressing this challenge and multiple studies demonstrate that AI can be a key enabling technology and may support industrial companies with complex problems [7].
3. In general, it is possible that AI is also applied in the product recycling stage, which is an important part of a circular economy. For instance, it has been pointed out that AI may support to loops by improving logistics chains in reverse logistics operations [10].
4. AI also can be used as a tool to support product design engineers during the design of the system architecture and product components. This is a bit less obvious than especially points one and two but the improvement potential is very high. One prominent example is design optimization; in this endeavor AI can help to avoid getting stuck in local minima or maxima and to explore the design space efficiently for finding the global optima [11].
5. Similarly, the production system engineers can also be supported during the design of the production system architecture and the design of the production tools and the selection of assembly resources. Also, the generation of the computer programs e. g. for PLCs or of energy efficient robot trajectories can be AI supported [9].
6. In the product development V-model, verification is an important activity which can also be supported by AI. For instance, AI can help to develop testing plans; engineer may more efficiently search through the space of possible experiments and find the global optima much faster than with conventional approaches [11].
7. Also, during the verification of the production system AI can be applied in a similar manner than in the verification of the product (point 6 in Fig. 1).

The application of AI in all seven stages of an integrated design process requires a considerable amount of data which contains structural information concerning the product and production systems as well as procedural data concerning parameters and behavior of all entities; the complete structural and procedural information over the whole life-cycle can form a digital twin (DT), which also enable bilateral data exchange with the real world [9]. Additionally, for a re-integration of the information generated by the AI, a modular approach is necessary. The scientific approach described in this paper is based on the one hand on GBDLs, which are used as the basis of an engineering framework using the Unified Modeling

Language (UML) which is implemented in the software Design Compiler 43™ (DC43) and on the other hand on functional mock-up units (FMUs) which are applied in order to enable a modularized co-simulation framework for the integrated development process.

It is important to note that international standards and frameworks have to be considered in this endeavor, notably the Industrial Ontologies Foundry (IOF) [12], the Industrial Data Ontology (IDO) [13], the Information Modelling Framework (IMF) [14] and the Industrial Digital Twin Association (IDTA) [15]; for a concise overview see [16]. However, the work in this paper is still in its development phase and it still has to be determined which information entities are part of a mandatory information exchange and should therefore be part of a future consolidated standard and framework or just optional. As an analogy, the creation of the UML standard [17] was the final synthesis of three already proven methods for object oriented modelling that were already implemented and successfully executed in a multitude of software and engineering projects.

This paper aims to present a concept of a framework together with its component and the application to the design of automated assembly system; this aim leads to the following structure of the paper. Section 2 described the concept of a self-learning twin. The sensible application of GBDLs and the underlying ontologies for knowledge representation are described in Section 3. The use of FMUs in the integrated product development process and the modular integration of AI are described in Section 4. Section 5 is focused on the description of the application to the automated assembly system, while Section 6 gives a summary and outlook.

## 2. Concept of a self-learning digital twin

The central aim of the underlying research is to design an AI-based, self-learning and self-explanatory digital twin [18] that automatically adapts to the real system behavior and represents a consistent image of the production process and product life cycle at all times. A central, semantically rich model of the product, the production process and the product life cycle serves as the basis for a sensible application of AI. Insights gained using AI methods are in turn integrated into the overall model and are thus brought into an overall context. Such an overall model improves the interpretability and explainability of the AI models as well as their automatic adaptation and enables complex analyses and forecasts, in particular by means of simulation techniques (e. g. the prediction of expected performance and prevention of faults or the long-term optimization of the production process). The core element of the approach is a novel, AI-based, self-learning engineering model consisting of a product and production model that maps function, behavior and product geometry (Figure 2).

In the background the real production system and the real product are visible, containing amongst others sensors, product planning systems (PPS) and further external data sources. The self-learning digital twin, which is bidirectionally connected with the real world, contains the product, production and behavior models, the simulation environment as well as the AI components. The framework based on the concept shown in Figure 2 is suitable for the integrated, lifecycle-spanning mapping of cross-domain product and process data in the areas of function, behavior, structure and geometry. Sensors (e.g. including 3D sensors and camera technology) and actuators (e.g. integrated via a production planning and control system) are used to collect performance and execution data from the production process and link it to the product and production model in the form of a semantic annotation. AI techniques from the field of machine learning (e. g. artificial neural networks [ANNs] and deep learning) are used to evaluate the performance and execution data with the aid of the semantic model and generate insights into cause-effect relationships and process patterns in the form of AI models.

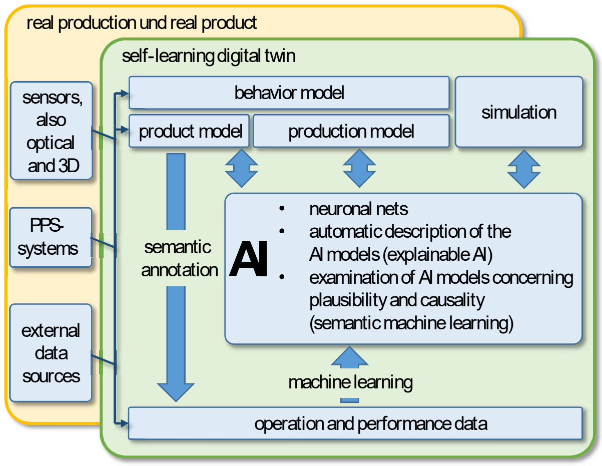


Fig. 2. Concept of a self-learning AI-based digital twin

These are then integrated into the overall model in the sense of a self-learning digital twin and serve as input for the simulation of product [19], [20]. This type of successive enrichment of the overall model with recognized patterns and correlations allows the comparison with empirical values of previous findings and the (semi-)automatic testing of the findings for plausibility and causality [21], [22]. Techniques from the field of XAI are used to automatically explain and describe the AI models [23], e. g. by characterizing influential features or layer-wise relevance propagation. A methodological focus here is on the conception and testing of novel approaches for building understandable AI models based on a semantically rich product and production model (e. g. the construction of decision trees based on the overall model, which maps function, behavior and product geometry), especially as surrogate models for black box models such as ANNs [24].

During the initial phases of the research the abstract concept of a self-learning digital twin was transformed into a technological blue-print, which is shown in Figure 3 – based on [25]. The structural and procedural information concerning the product and the production system are generated using both the ontologies, vocabulary and rules based on GBDLs (compare Section 3) and product and production operation data (measurements, parameters). This information is stored in graph-based form in a consistent knowledge graph. The unique quality of the application of the design compiler integrated in DC43 is the capability to create 3D geometry information from this knowledge graph. The knowledge graph also allows to build simulations using FMUs achieving a modular co-simulation environment for complex production system operations. Additionally, from the knowledge graph training data for machine learning and XAI can be generated and the resulting models can also be integrated in from of FMUs in the co-simulation environment. It is apparent that GBDLs are an essential element of the blueprint; some main characteristics and the possibility to represent knowledge in ontologies is the focus of the next section.

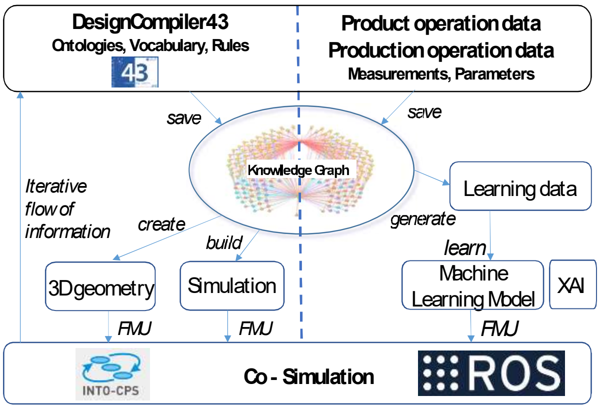


Fig. 3. Blueprint of the digital twin

## 3. GBDLs – ontologies for knowledge representation

Since several years, engineering frameworks based on GBDLs are in the center of large-scale research projects [26], [27] and have found several applications in industry. For a detailed description, the publications [28], [29], and [30] can be consulted. In this paper, the discussions of GBDLs is concentration on the application of ontologies for knowledge representation. To clarify their contribution, the general information flow in graph-based design languages is shown in Figure 4 compare [31], [27].

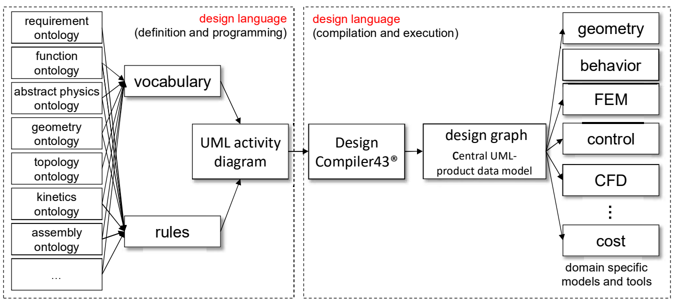


Fig. 4. Information flow in GBDLs (compare [31], [27])

The central entities of a GBDL are the vocabulary (e.g. the components of a product) and the rules (e.g. information how components can be incorporated in a product). These entities can be represented using UML and the information can be stored in a central knowledge graph (compare also Figure 3). The information content of a GBDL for a certain product family is enormous; an efficient application of GBDLs requires the availability of engineering knowledge in a sensible and pre-structured form – this can be achieved by employing ontologies. In general, ontologies dispose a representation vocabulary providing a set of terms to describe entities in a specific domain; and a body of knowledge providing a set of facts associated with conceptual models [32], [33]. Through this, an ontology specifies the characteristics of models, makes them explicit and can be referred to as knowledge models [34]. Ontologies arrange, classify and categorize (compare [33]) entities and their relationships and may include axioms (widely accepted principles which are fundamental concepts of certain entities or processes [35]: In engineering frameworks based on GBDLs, these ontologies concern different domains:

- **The requirement domain**: the central entity in this domain are requirements. Requirements can belong to certain categories (e. g. functional requirements vs. property requirements), can be part of a requirement hierarchy and can be associated to modules, components and features of the technical system (compare [27]).
- **The functional domain**: the central entity in this domain are processes which realize certain functional requirements. Another very important entity are operands in form of matter, energy and signal. Important associations concern actors, use cases and stakeholders (compare [36]).
- **The abstract physics domain**: the central entity in this domain are the physical and logical phenomena which realize the processes as well as the static and dynamic behavior of modules, components and features of the technical system.
- **The geometry domain**: the central entity is the geometry of modules, components and features of the technical system. The geometrical features inherit from abstract geometrical features (e. g. cylinder) and allow volume addition and subtraction operations.
- **The technology domain**: the central entity are technological realizations of physical and logical phenomena. Established technologies and innovations can be represented.
- **The kinetics domain**: this domain is a specialization of the abstract physics domain concentrating on movements of components, joint and degrees of freedom as well as forces, moments, stresses and deformations (compare CIRP KIDZ). These entities can also be the structural input information for FMUs in a co-simulation environment [37].
- **The assembly domain**: the central entities are assembly processes, assembly resources and assembly layout considerations. This domain is the special focus of later parts of this paper.

In the blueprint of the self-learning DT (compare Figure 3) FMUs play a central role for the seamless integration of AI in simulation processes. This aspect is the focus of the next section.

## 4. FMUs – modular integration of AI

This section is focusing on a modular simulation and co-simulation environment as part of a self-learning DT and the seamless integration of AI models in this environment. A promising approach for simulation modularization is the application of FMUs, which are defined in the functional mock-up interface (FMI) standard developed by the Modelica association [4], [37]. The FMI standard provides a standardized interface and allows the exchange and encoding of simulation data. The complete system can be decomposed into sub-models, which can separately be simulated. One exemplary advantage in kinematic applications is the fact that mechanically rigid couplings can be avoided which may lead to unwanted oscillations. Essentially, a FMU may be realized as a file directory which contains a "modelDescription.xml" – file. This file describes the input and output of the FMU and what type of parameter are used. The main advantage of Co-Simulation with FMUs instead of a monolithic simulation is the flexibility to use different simulations tools such as but also to incorporate AI and XAI models. The main aspects of this approach are shown in Figure 5.

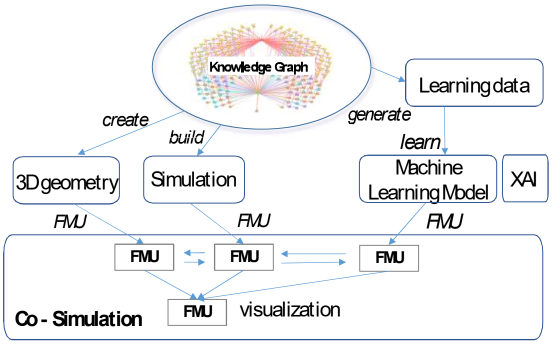


Fig. 5. Modular co-simulation environment with XAI model(s)

It is important to note that not only simulation tools (such as ANSYS™ or OpenFoam™) and AI/XAI models can be connected without changing the simulation architecture, but also hardware-in-the-loop (HiL) systems. By means of the integration of AI/XAI into the co-simulation environment via FMUs, the opportunity arises to use AI in all domains or intended simulations; design engineers gain the flexibility to alternate between conventional physical simulations and AI based FMUs. Also, the visualization of simulation results can be part of the co-simulation environment, e.g. using the software Unity™ shown in Figure 6.

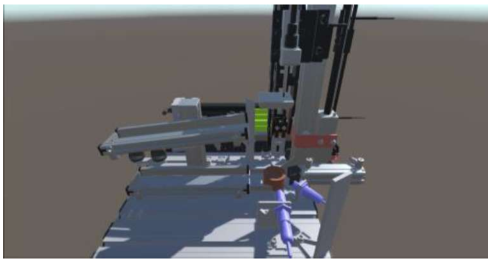

Fig. 6. Geometry representation in Unity™

The sample application is a Festo™ automated assembly plant, which contains a slide for a product component which applies air pressure for friction reduction. The friction behavior in this kind of situation cannot easily be modelled. It was possible to learn an accurate machine learning model and to develop an XAI decision tree. Figure 6 shows this example as Unity™ visualization, the simulation of a slide of a Festo automated assembly system™.

In Unity, the visualization of static and dynamic components can be performed and collision occurrence and characteristics can be defined. In fact, the entire scene in Fig. 6 is automatically generated via translation of GBDLs inside DC43 and serves as an example of such an automated process chain.

## 5. Application of the framework

One of the cornerstones of the proposed approach is the storage of product, production sequence and production system in one consistent model together with product operation data as well as production operation data. The integrated development framework that enables this, is shown in Figure 7.

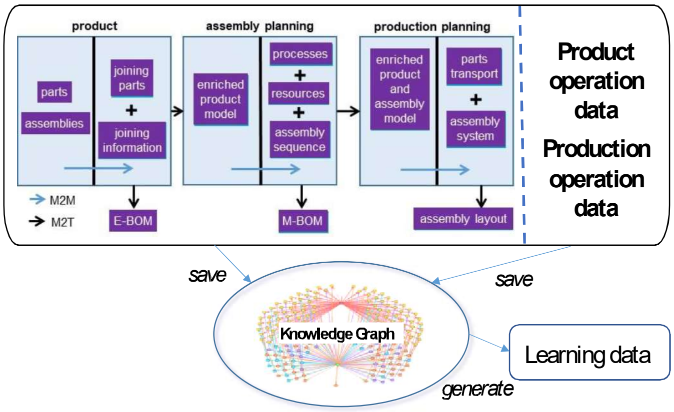


Fig. 7. Integrated development framework

It is a well-known fact, that data need reference (meaning) to become information and that information needs context in order to become knowledge. A knowledge graph as visible in Figure 7, which can serve for AI implementation, needs to contain product operation data and production operations, but also the reference entities such as product sub-modules or assembly processes. These entities need to be connected in a sensible manner to describe the context.

In the upper part of Figure 7, the essential model-to-model transformation are visible, which allow a seamless connection of product and production entities. From the product structure with parts and assemblies the assembly sequence as well as the assembly system is automatically generated; one example result is shown in Figure 8.

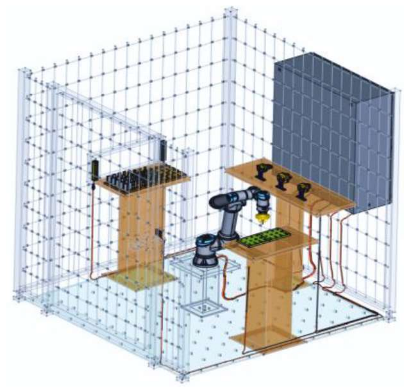

Fig. 8. Result of the framework application

The Figures. 6, 7, and 8 illustrate the generative power of the automated information flow in GBDLs as explained in Section 3, for which the design graph serves as the knowledge graph as the central repository of all design and production knowledge which can be mapped in arbitrary target formats.

## 6. Conclusions and outlook

The main intention of the research described in this paper is the development of a framework which allows the application of AI in MBSE development processes of products and the respective production systems. Initially the concept o a self-learning DT was explained; this DT combines product, production system, simulation, and operation information in order to enable AI/XAI application. The means to achieve the information combination in this DT are GBDLs; such GBDLs lead to the storage of all relevant information with the context in a consistent central data model – the knowledge graph. Another main challenge in the DT application is the simulation of dynamic processes. In the proposed concept of the self-learning DT, FMUs enable a modular simulation environment. FMUs can be AI based; such AI-FMUs can be seamlessly integrated in the simulation environment and can interact with conventional FMUs based on e. g. differential equations describing physical relationships.

## Acknowledgements

Parts of the described research were funded in by the Carl Zeiss Foundation in the scope of the project AI-based digital twin (KI-basierter digitaler Zwilling - KIDZ).

## References

[1] Peretz-Andersson, E., Tabares, S., Mikalef, P., & Parida, V. (2024). Artificial intelligence implementation in manufacturing SMEs: A resource orchestration approach. International Journal of Information Management, 77, 102781.

[2] Demlehner, Q., Schoemer, D., & Laumer, S. (2021). How can artificial intelligence enhance car manufacturing? A Delphi study-based identification and assessment of general use cases. International Journal of Information Management, 58, 102317.

[3] IILS mbH. Design Complier 43™. (2024) Online: https://www.iils.de (accessed on 15 October 2024).

[4] Miller, M., Pfeil, M., & Kennel, R. (2023). Trailer Electrification–A HIL Approach for MPC Powertrain Control to Ensure Driver Safety in Micromobility (No. 2023-24-0180). SAE Technical Paper.

[5] VDI/VDE 2206: Entwicklung cyber-physischer mechatronischer Systeme (CPMS). Beuth: 2020.

[6] Rana, K., & Khatri, N. (2024). Automotive intelligence: Unleashing the potential of AI beyond advance driver assisting system, a comprehensive review. Computers and Electrical Engineering, 117, 109237.

[7] Leberruyer, N., Bruch, J., Ahlskog, M., & Afshar, S. (2023). Toward Zero Defect Manufacturing with the support of Artificial Intelligence—Insights from an industrial application. Computers in Industry, 147, 103877.

[8] Abbasi, S., & Rahmani, A. M. (2023). Artificial intelligence and software modeling approaches in autonomous vehicles for safety management: a systematic review. Information, 14(10), 555.

[9] Olatunde, T. M., Okwandu, A. C., Akande, D. O., & Sikhakhane, Z. Q. (2024). Reviewing the role of artificial intelligence in energy efficiency optimization. Engineering Science & Technology Journal, 5(4), 1243-1256.

[10] Ghoreishi, M., & Happonen, A. (2020, May). Key enablers for deploying artificial intelligence for circular economy embracing sustainable product design: Three case studies. In AIP conference proceedings (Vol. 2233, No. 1). AIP Publishing.

[11] Gutierrez, D. P., Folkmann, L. M., Tribukait, H., & Roch, L. M. (2023). How to accelerate R&D and optimize experiment planning with machine learning and data science. Chimia, 77(1/2), 7-16.

[12] https://oagi.org/pages/industrial-ontologies. Accessed on 03.02.2025.

[13] https://rds.posccaesar.org/ontology/lis14/. Accessed on 03.02.2025.

[14] https://www.dnv.com/digital-trust/recommended-practices/asset-information-modelling-dnv-rp-0670/. Accessed on 03.02.2025.

[15] https://industrialdigitaltwin.org/en/technology. Acc. on 03.02.2025.

[16] Lyu, Z. Handbook of Digital Twins. CRC Press: 2024.

[17] http://www.uml.org/. Accessed on 03.02.2025.

[18] Trauer, J., Schweigert-Recksiek, S., Engel, C., Spreitzer, K., & Zimmermann, M. (2020, May). What is a digital twin?–definitions and insights from an industrial case study in technical product development. In Proceedings of the design society: DESIGN conference (Vol. 1, pp. 757-766). Cambridge University Press.

[19] Publio, G. C., Esteves, D., Ławrynowicz, A., Panov, P., Soldatova, L., Soru, T., ... & Zafar, H. (2018). ML-schema: exposing the semantics of machine learning with schemas and ontologies. arXiv preprint arXiv:1807.05351.

[20] McGuinness, D. L., Ding, L., Da Silva, P. P., & Chang, C. (2007, July). PML 2: A Modular Explanation Interlingua. In ExaCt (pp. 49-55).

[21] Chen, J., Lécué, F., Pan, J. Z., Horrocks, I., & Chen, H. (2018, September). Knowledge-based transfer learning explanation. In Sixteenth International Conference on Principles of Knowledge Representation and Reasoning.

[22] Mahajan, D., Tan, C., & Sharma, A. (2019). Preserving causal constraints in counterfactual explanations for machine learning classifiers. arXiv preprint arXiv:1912.03277.

[23] Geng, Y., Chen, J., Jiménez-Ruiz, E., & Chen, H. (2019). Human-centric transfer learning explanation via knowledge graph. arXiv preprint arXiv:1901.08547.

[24] Confalonieri, R., del Prado, F. M., Agramunt, S., Malagarriga, D., Faggion, D., Weyde, T., & Besold, T. R. (2019). An ontology-based approach to explaining artificial neural networks. arXiv preprint arXiv:1906.08362.

[25] Arff, B., Haasis, J., Thomas, J., Bonenberger, C., Höpken, W., & Stetter, R. (2023). Analysis and visualization of production bottlenecks as part of a digital twin in industrial IoT. Applied Sciences, 13(6), 3525.

[26] Rudolph, S. A Semantic Validation Scheme for Graph-Based Engineering Design Grammars. In: Gero, John (ed): Proceedings Design Computing and Cognition (DCC'06), Springer, Dordrecht, 541–560, 2006.

[27] Holder, K., Zech, A., Ramsaier, M., Stetter, R., Niedermeier, H.-P., Rudolph, S. and Till, M. (2017) "Model-Based Requirements Management in Gear Systems Design based on Graph-Based Design Languages". Appl. Sci. 2017, 7, 1112, doi:10.3390/app7111112.

[28] Schopper, D., Tonhäuser, C. (2021). A User-friendly Planning Tool for Assembly Sequence Optimization. In Advances in Automotive Production Technology–Theory and Application: Stuttgart Conference on Automotive Production (SCAP2020) (pp. 28-36). Springer Berlin Heidelberg.

[29] Dinkelacker, J.; Kaiser, D.; Panzeri, M.; Parmentier, P.; Neumaier, M.; Schopper, C.; Rudolph, S;. (2023). System integration based on packing, piping and harness routing automation using graph-based design languages. CEAS Aeronautical Journal. 14. 10.1007/s13272-023-00642-5.

[30] Zech, A.; Stetter, R.; Till, M.; Rudolph, S.: Automated Generation of Clamping Concepts and Assembly Cells for Car Body Parts for the Digitalization of Automobile Production. Stuttgart Conference of Automotive Production SCAP 2020, 2nd to 10th November 2020.

[31] Rudolph, S. On the problem of multi-disciplinary system design—And a solution approach using graph-based design languages. In Proceedings of the 1st ACCMWorkshop on Mechatronic Design, Linz, Austria, 30 November 2012.

[32] Yang, L., Cormican, K., & Yu, M. (2019). Ontology-based systems engineering: A state-of-the-art review. *Computers in Industry*, *111*, 148-171.

[33] Arista, R., Zheng, X., Lu, J., & Mas, F. (2023). An Ontology-based Engineering system to support aircraft manufacturing system design. Journal of Manufacturing Systems, 68, 270-288.

[34] Darlington, M. J., & Culley, S. J. (2008). Investigating ontology development for engineering design support. Advanced Engineering Informatics, 22(1), 112-134.

[35] Kulak, O., Cebi, S., & Kahraman, C. (2010). Applications of axiomatic design principles: A literature review. Expert systems with applications, 37(9), 6705-6717.

[36] Elwert, M., Ramsaier, M., Eisenbart, B., Stetter, R., Till, M., & Rudolph, S. (2022). Digital function modeling in graph-based design languages. Applied Sciences, 12(11), 5301.

[37] Saft, P.; Pfeil, M.; Stetter, R.; Till, M.; Rudolph, S.: Integration of geometry modelling and behavior simulation based on graph-based design languages and functional mockup units. Procedia CIRP 2024.