Automatic Model Generation for Discrete-Event Simulation

Sanket Kute
Institute for Engineering of Products and Systems (IEPS), Otto von Guericke University Magdeburg, Germany
sanket.kute@ovgu.de

Sebastian Lang
Institute for Engineering of Products and Systems (IEPS), Otto von Guericke University Magdeburg, Germany

Abstract
Automatic model generation for manufacturing-oriented discrete-event simulation (DES) aims to reduce the effort required to build and update material flow and production system models. Established automatic simulation model generation (ASMG) approaches often rely on structured engineering data, event logs, or predefined transformation logic. Recent progress in large language models creates new opportunities to use less formalized information, including textual planning notes, layout descriptions, process assumptions, and parameter tables. However, current LLM-assisted approaches still face limitations in reliability, transparency, reproducibility, and simulator integration. This paper derives a conceptual framework for LLM-assisted automatic model generation in manufacturing-oriented discrete-event simulation. The proposed workflow combines hybrid information extraction, a simulation-oriented intermediate representation, explicit validation, simulator-specific instantiation, and expert review. The LLM is positioned as a controlled component for semantic interpretation and structuring, not as an autonomous end-to-end model generator. Particular attention is given to behavioural and control-related aspects because these are required for executable and useful manufacturing simulation models. The paper concludes by outlining a phased prototype implementation and evaluation strategy using controlled manufacturing use cases and later validation with an industrial factory-planning partner.

1. Introduction
Discrete-event simulation (DES) is widely used in manufacturing planning to analyse material flows, resource utilization, routing decisions, and dynamic system behaviour before changes are implemented in the real system. Despite this value, DES models still require substantial effort for data preparation, conceptual modelling, implementation, and validation. In industrial practice, this effort becomes especially relevant when production systems change, but existing simulation models are not updated with the same speed. This gap has long motivated research on automatic model generation, which aims to reduce modelling effort and improve the timeliness of simulation-based decision support in manufacturing (Cimino et al., 2024; Reinhardt et al., 2019).

1.1. Established foundations of automatic simulation model generation
Automatic simulation model generation (ASMG) is not a new research topic. Earlier work already showed that simulation models can be derived from structured or semi-structured information sources such as ERP data, engineering artefacts, plant layouts, event logs, and PLC-related information. In this body of literature, the central challenge is usually not the interpretation of free-form input, but the transformation of available system information into simulator-ready model structures. Bergmann and Strassburger (2010) therefore describe ASMG as a field shaped by parametric, structural, and hybrid knowledge-based approaches, each of which depends strongly on the type, completeness, and granularity of the available source data.

The literature also shows that ASMG is already an established field, even if it is not yet fully unified conceptually. Reinhardt et al. (2019), in their survey on material flow simulation in discrete manufacturing, show that ASMG has been applied in a broad range of industrial contexts, including production planning, bottleneck analysis, layout design, and virtual commissioning. Their review is particularly useful because it organizes the field by data source, data variability, and information retrieval rather than by one dominant modelling method. Cimino et al. (2024) reach a similar conclusion in their broader systematic review and note that the field still contains overlapping categories and partly inconsistent definitions. Taken together, these studies indicate that ASMG is

http://dx.doi.org/10.25673/123555

already a substantial body of work, but not yet a fully consolidated methodological discipline.

1.2. Structured and data-driven ASMG approaches

A large part of the more mature ASMG literature is based on structured engineering artefacts and operational data. Oppelt et al. (2014) generate simulation models for virtual commissioning from plant engineering data. Arroyo et al. (2016) derive qualitative simulation models from legacy P&IDs by combining optical recognition and semantic analysis. Santillán Martínez et al. (2018) further show that 3D plant models can provide geometric and spatial information relevant for higher-fidelity process simulation. These studies point in the same direction: automatic model generation becomes more feasible when relevant system knowledge is already available in engineering artefacts that can be systematically mapped into simulation structures.

A related research stream is based on operational and event data. May et al. (2024) infer routing, control, and temporal aspects of material flow simulations from event-based production data using process mining and machine learning. Lugaresi and Matta (2023) likewise show the potential of event-based derivation, but also point out that more complex assembly systems require richer event semantics than standard traces usually provide. Tan et al. (2023) extend this line of work by combining automatic model generation with data assimilation so that the simulation model can adapt when the real system changes. These approaches show that data-driven ASMG can be effective where suitable production data is available, but they also remain dependent on already existing or observable systems.

Across both streams, intermediate representations play an important methodological role. Lütjen and Rippel (2015) treat simulation generation as a transformation problem based on a factory data model. Pagliares and Hirata (2018) use an intermediate XML-based notation to map process specifications into simulation diagrams. Krenczyk and Bocewicz (2015) likewise rely on a neutral intermediate model in the context of ERP-DES integration. This recurring use of representation layers is important because it shows that successful ASMG has long depended on structured transformations between source information and executable simulator models rather than on direct one-step conversion.

1.3. Emerging LLM-assisted simulation model generation

Recent progress in large language models has introduced a new perspective on this problem. Instead of relying only on already structured engineering data, current research investigates whether simulation-relevant model content can also be derived from heterogeneous system information that may include textual descriptions, semi-structured artefacts, and structured planning data. This is especially relevant in early planning contexts, where layout ideas, process descriptions, assumptions, and constraints are often available before they are fully formalised.

Current studies also show that LLM-assisted generation is still an emerging direction rather than a mature replacement for established ASMG methods. Jackson et al. (2024) show that simulation models for queuing and inventory systems can be generated from descriptive input with functionally valid results, but the scope remains limited to comparatively simple logistics-oriented cases. Carreira-Munich et al. (2024), by contrast, propose a more structured approach in DEVS Copilot, where semantic understanding is separated from syntactic code generation through an intermediate formal layer. Their work is particularly relevant because it shows that current LLMs are promising assistants, but not yet reliable enough for fully autonomous use in settings where correctness is critical.

The same need for structure is visible in manufacturing-related work. Hu et al. (2025) address shopfloor layout generation through a structured metamodel, JSON-based parsing, and an iterative think-generate-evaluate cycle. In our previous work, Kute et al. (2025) presented a feasibility study for generating Plant Simulation models from layout-oriented system descriptions through structured output, validation, and interpreter-based instantiation. At the same time, that study also highlighted open issues such as verbose representations, token limitations, and spatial correctness. These recent contributions point in a consistent direction. LLMs may widen the range of usable input forms and improve accessibility, but they do not remove the need for structured representations, validation logic, and simulator-specific transformation layers.

1.4. Research gap and contribution of this paper

The reviewed literature points to three main observations for the present paper. First, ASMG is an established research field, but many mature approaches rely on structured engineering data, event logs, or formal transformation logic. Second, successful approaches rarely generate executable models in one step. They usually depend on intermediate representations, staged transformations, and validation mechanisms. Third, LLM-assisted model generation is promising, but still limited by reliability, reproducibility, and integration into industrial simulation workflows.

The relevant gap is therefore not the general use of AI in ASMG. The open issue is how LLM-based interpretation can be connected with the structured logic of simulation model generation. Current literature does not yet sufficiently explain how heterogeneous manufacturing information can be harmonized, interpreted, represented, validated, and translated into simulator-specific artefacts in a transparent and reproducible workflow. This is particularly important for manufacturing-oriented DES because useful models require not only structural elements such as sources, stations, buffers, and sinks, but also routing rules, operating constraints, priorities, and other behavioural logic.

Against this background, this paper derives a conceptual framework for LLM-assisted automatic model generation in manufacturing-oriented DES, with focus on material flow and production system models. The contribution is conceptual rather than empirical. Its novelty does not lie in replacing established ASMG principles, but in embedding LLM-based semantic interpretation into a structured ASMG workflow. Compared with classical transformation- and metamodel-based approaches, the proposed framework addresses an earlier and less formalized modelling stage, where relevant system information may be distributed across layout data, planning documents, process descriptions, assumptions, and parameter tables. Particular attention is given to process behaviour and control logic because these aspects remain difficult to derive from heterogeneous early-stage information.

2. Research Design

This paper follows a conceptual and design-oriented research approach. It does not report a completed implementation or empirical evaluation. Instead, it develops a methodological basis for LLM-assisted ASMG in manufacturing-oriented DES. The analysis builds on two related literature streams: established ASMG research and recent work on LLM-assisted simulation generation. The purpose is to identify which methodological principles remain necessary when heterogeneous system information, rather than only formalized engineering data, is introduced into the model-generation workflow.

The research design follows a staged modelling logic. Heterogeneous system information is first interpreted and harmonized, then represented in a structured form, checked for consistency, and finally prepared for simulator-specific instantiation. This reflects a central principle from both established ASMG research and recent LLM-based work: reliable model generation requires explicit representation layers, structured transformations, and validation mechanisms.

Based on this literature-based synthesis, Section 3 presents the proposed framework as the conceptual result of the paper. The framework is intended as a basis for later implementation and evaluation, not as a validated operational system.

The planned follow-up work will operationalize the framework through a prototype-based research design. In a first step, a fixed LLM version will be selected and used only for defined extraction and structuring tasks. The initial prototype will use an instruction-following LLM with support for structured JSON output. The model will not generate an executable simulation model directly. Instead, it will extract simulation-relevant information and represent it according to a predefined intermediate representation. This representation will be specified as a JSON-based schema with structural, parametric, and behaviour-related fields. The prompting strategy will follow a staged procedure in which the LLM first identifies relevant entities and relations, then completes parameters and assumptions, and finally produces schema-compliant output. Reproducibility will be supported by storing the model version, prompts, parameter settings, input cases, generated intermediate representations, validation reports, and resulting Plant Simulation artefacts.

3. Conceptual Framework and Discussion

3.1. Proposed Conceptual Framework

Figure 1 presents the conceptual framework proposed in this paper for LLM-assisted automatic model generation in manufacturing-oriented discrete-event simulation. Since the paper is conceptual in scope, this section does not report empirical performance results. Instead, it presents the derived framework as the main conceptual result of the literature-based research design. The framework organizes model generation as a staged workflow in which heterogeneous manufacturing system information is extracted, structured, validated, and then translated into a simulator-specific model.

At the input level, the framework starts from heterogeneous input sources. In contrast to approaches that assume only textual input, the proposed framework takes a broader view of available system information. This may include layout and topology data, resource and property data, process, routing and timing data, textual requirements and assumptions, and existing planning or engineering artefacts. Such a broader input perspective is necessary because manufacturing system descriptions are rarely available in one uniform format. In practice, relevant modelling information is usually distributed across structured, semi-structured, and

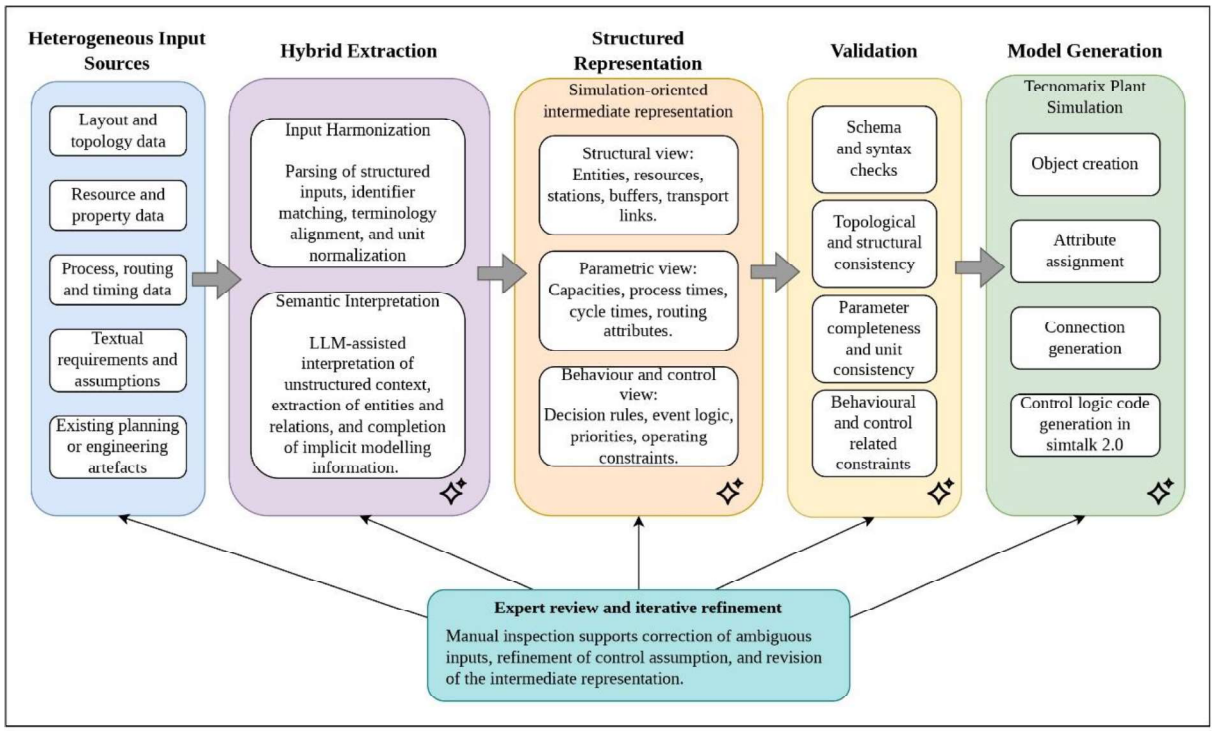

Figure 1: Conceptual framework for LLM-assisted automatic simulation model generation.

unstructured sources that must first be brought into a usable form.

Once these inputs are available, the framework moves to hybrid extraction. This stage combines input harmonization and semantic interpretation. Input harmonization refers to the parsing of structured inputs, identifier matching, terminology alignment, and unit normalization. Semantic interpretation addresses those parts of the input that are less formalized and uses the LLM to identify simulation-relevant entities, relations, and implicit modelling assumptions. The LLM is therefore not positioned as an autonomous model generator, but as one component within a broader extraction and interpretation process. This design decision reflects the current state of research, where direct one-shot generation still remains difficult when correctness, transparency, and simulator integration are required.

At the core of the framework lies the simulation-oriented intermediate representation. This stage separates input interpretation from simulator-specific implementation and organizes the extracted system information into a structured representation that is independent of the final simulator. In the proposed framework, this representation consists of three complementary views. The structural view contains entities such as resources, stations, buffers, and transport links. The parametric view contains model attributes such as capacities, process times, cycle times, and routing parameters. The behaviour and control view contains decision rules, event logic, priorities, and operating constraints. This separation is methodologically useful because it allows structural model content and behavioural logic to be represented together within one formal layer rather than treating them as separate modelling problems.

Before any simulator-specific instantiation takes place, the framework introduces validation and consistency checking as an explicit stage of the workflow. In the planned implementation, validation will be separated into several targets. First, extraction correctness will be checked by comparing the entities, connections, parameters, and control rules identified by the LLM with the information contained in the input sources. Second, the intermediate representation will be checked for schema validity, completeness, allowed value ranges, unit consistency, and topological consistency. Third, the generated simulator model will be checked for executability in Tecnomatix Plant Simulation, including object creation, connection generation, attribute assignment, and control-logic integration. Fourth, the behaviour and performance of the generated model will be assessed through use-case-specific validation criteria. These criteria may include correct routing behaviour, expected bottleneck locations, throughput, lead time, work in process, queue or buffer levels, and resource utilization. Where reference data or manually created models are available, these indicators can be compared against them. Where such references are not available, expert review can be used to assess whether the generated model is plausible for the intended planning task. Finally, the reduction of modelling effort will be evaluated through the number of required manual corrections and the time needed to obtain an executable and acceptable model.

Only after successful validation does the workflow proceed to simulator-specific model generation. In the present framework, this stage is represented by Tecnomatix Plant Simulation. The validated intermediate representation is translated into simulator-specific artefacts such as object creation, attribute assignment, connection generation, and control-logic implementation in SimTalk 2.0. In the planned prototype, two implementation paths will be considered. The first path is interface-based instantiation, where the model is generated through available Plant Simulation interfaces or SimTalk scripts. The second path is file-based instantiation using the Plant Simulation Folder Model format. In this format, selected model artefacts can be stored as structured YAML files, which may support version control, inspection, and reproducibility. This file-based option does not replace the intermediate representation. Instead, it is treated as a Plant Simulation-specific target format that is generated only after the intermediate representation has passed validation.

A final element of the framework is expert review and iterative refinement. At the current stage of research, LLM-assisted ASMG should be understood as supporting modelling work rather than fully replacing expert judgement. Manual review remains necessary for resolving ambiguous input, refining modelling assumptions, and revising the intermediate representation where inconsistencies are detected. The feedback loops shown in Figure 1 therefore indicate that the process is iterative and that human intervention remains an integral part of the proposed workflow.

3.2. Discussion of the Framework

The proposed framework contributes to the identified research gap by extending the structured logic of classical ASMG toward heterogeneous and less formalized manufacturing system information. Existing transformation- and metamodel-based approaches are strong when the required model information is already available in a formal or semi-formal structure. However, early factory planning often contains information that is incomplete, textual, assumption-based, or distributed across different artefacts. This paper addresses this earlier modelling stage by using the LLM for semantic interpretation and information structuring, while the subsequent model generation remains controlled through an intermediate representation, validation, and simulator-specific transformation. In this sense, the LLM is not treated as a replacement for ASMG methods, but as a controlled front-end component that can widen the range of usable input information.

Two initial manufacturing use cases are suitable for making the proposed framework operational. The first use case is the generation of an early material flow model for a small production or assembly system. The input may consist of a preliminary layout, a list of stations and buffers, process descriptions, cycle times, buffer capacities, routing assumptions, and textual planning notes. The expected output is an executable Plant Simulation model that represents the main stations, material flow connections, processing times, buffer logic, and basic routing behaviour. This use case is suitable for evaluating whether the framework can transform incomplete early-stage planning information into a first simulation model for layout and bottleneck analysis. The second use case is the update of an existing material flow model after changes in layout, parameters, or routing logic. In this case, the input may consist of an existing model description, updated process tables, change notes, and revised planning assumptions. The expected output is an updated intermediate representation and a regenerated or modified Plant Simulation model. This use case is suitable for evaluating whether the framework can reduce the effort required to keep simulation models consistent with planning changes.

A particularly important aspect of the framework is that it includes not only structural model elements, but also behaviour and control-related aspects. This is relevant because manufacturing-oriented discrete-event simulation usually requires more than the placement of stations, buffers, and connections. It also requires the formalization of routing logic, decision rules, operating constraints, and other behavioural mechanisms that influence system performance. By assigning these elements to the intermediate representation and by subjecting them to explicit validation, the framework provides a clearer methodological basis for later implementation and evaluation.

At the same time, the framework should be understood as a conceptual research result, not as a validated operational system. Its contribution in the present paper lies in the structured organization of the modelling workflow and in the explicit definition of the interfaces between input processing, representation, validation, and simulator-specific generation. This creates a coherent basis for the later development of implementation artefacts and for the future assessment of model validity, manual correction effort, reproducibility, and practical usefulness in manufacturing simulation contexts.

4. Limitations and Conclusion

4.1. Limitations

The present paper has several limitations. First, the contribution is conceptual. The framework has been derived from the literature and from methodological requirements of manufacturing-oriented DES, but it has not yet been implemented or systematically evaluated. The paper therefore cannot provide

quantitative evidence on model quality, modelling effort, reproducibility, or robustness in practical application. This limitation follows from the present scope, which is to define a structured research direction rather than report completed implementation results.
Second, the framework assumes that relevant system information can be made available in a form that is sufficiently interpretable for structured extraction and transformation. In practice, however, manufacturing information is often incomplete, inconsistent, distributed across different artefacts, or only partly formalized. This means that the quality of the generated model will depend strongly on the quality, completeness, and compatibility of the available input sources. The proposed workflow addresses this challenge through harmonization, intermediate representation, and validation, but these mechanisms do not eliminate the underlying data problem.

Third, the present framework is currently oriented toward manufacturing-related discrete-event simulation and is illustrated with Tecnomatix Plant Simulation as the target environment. This provides practical focus, but it also limits the immediate generalizability of the approach. Other simulation paradigms, broader supply chain settings, or alternative simulation tools may require different representation layers, transformation logic, or validation criteria. The same applies to control logic. Although the framework explicitly includes behavioural and control-related aspects, their complete and reliable formalization remains an open challenge, especially when rules are implicit, context-dependent, or only partially specified in the source information.

Finally, the role of the LLM within the framework should be interpreted carefully. The proposed approach does not assume that the LLM autonomously produces a complete and correct simulation model without human intervention. Instead, expert review remains necessary for resolving ambiguities, checking modelling assumptions, and refining the intermediate representation. For this reason, the framework should currently be understood as an approach for supported model generation rather than full modelling automation.

4.2. Conclusion and Outlook

This paper contributes a conceptual framework for LLM-assisted automatic model generation in manufacturing-oriented discrete-event simulation, with focus on material flow and production system models. The framework addresses the gap between heterogeneous early-stage manufacturing information and structured simulation model generation. Its main contribution is to position LLM-based semantic interpretation within a controlled ASMG workflow rather than treating the LLM as an autonomous model generator.

The proposed workflow combines flexible interpretation of heterogeneous input sources with a simulation-oriented intermediate representation, explicit validation, simulator-specific instantiation, and expert review. This structure provides a basis for later implementation in Tecnomatix Plant Simulation while keeping the generation process transparent and verifiable.

The next step in the research project is the implementation of the proposed framework in a prototype workflow. This work will follow a phased procedure. First, the intermediate representation will be specified as a JSON-based schema for structural, parametric, and behaviour-related model information. Second, a fixed LLM version and a documented prompting procedure will be used to extract and structure information from selected manufacturing planning inputs. Third, the generated intermediate representations will be checked for schema validity, topological consistency, parameter completeness, unit consistency, and basic control-logic constraints. Fourth, validated model specifications will be instantiated in Tecnomatix Plant Simulation through simulator-specific generation mechanisms. Finally, the generated models will be evaluated using use-case-specific validation criteria. The evaluation will consider extraction correctness, simulator executability, behavioural and performance plausibility, manual correction effort, reproducibility, and practical usefulness compared with conventional manual modelling. Where reference models or real system data are available, selected indicators such as throughput, lead time, work in process, buffer levels, and resource utilization can be compared against them.

Initial validation will be conducted in a controlled setting before broader industrial application. A university lab demonstrator or documented teaching factory scenario can be used for the first prototype because the layout, process sequence, parameters, and expected system behaviour can be defined and checked systematically. In a later step, the framework will be evaluated together with an industrial factory-planning partner. This will make it possible to assess the approach under more realistic planning conditions, where relevant information is distributed across layout data, process descriptions, planning documents, parameter tables, and expert assumptions.

References

Arroyo, E., Hoernicke, M., Rodríguez, P., & Fay, A. (2016). Automatic derivation of qualitative plant simulation models from legacy piping and instrumentation diagrams. *Computers & Chemical Engineering*, *92*, 112–132.

https://doi.org/10.1016/j.compchemeng.2016.04.040

Bergmann, S., & Strassburger, S. (2010). Challenges for the automatic generation of simulation models for production systems. In *Proceedings of the 2010 Summer Computer Simulation Conference* (pp. 545–549). https://doi.org/10.5555/1999416.1999486

Carreira-Munich, T., Paz-Marcolla, V., & Castro, R. (2024). DEVS Copilot: Towards generative AI-assisted formal simulation modelling based on large language models. In *2024 Winter Simulation Conference (WSC)* (pp. 2785–2796). IEEE. https://doi.org/10.1109/WSC63780.2024.10838994

Cimino, A., Longo, F., Nicoletti, L., & Veltri, P. (2024). Automated simulation modeling: Ensuring resilience and flexibility in Industry 4.0 manufacturing systems. *Procedia Computer Science, 232*, 1011–1024. https://doi.org/10.1016/j.procs.2024.01.100

Hu, F., Wang, C., & Wu, X. (2025). Generative artificial intelligence-enabled facility layout design paradigm. *Applied Sciences*, *15*(10), Article 5697. https://doi.org/10.3390/app15105697

Jackson, I., Sáenz, M. J., & Ivanov, D. (2024). From natural language to simulations: Applying AI to automate simulation modelling of logistics systems. *International Journal of Production Research*, *62*(4), 1434–1457. https://doi.org/10.1080/00207543.2023.2276811

Krenczyk, D., & Bocewicz, G. (2015). Data-driven simulation model generation for ERP and DES systems integration. In K. Jackowski, R. Burduk, K. Walkowiak, M. Woźniak, & H. Yin (Eds.), *Intelligent Data Engineering and Automated Learning – IDEAL 2015.* (Lecture Notes in Computer Science, Vol. 9375, pp. 264–272). Springer. https://doi.org/10.1007/978-3-319-24834-9_32

Kute, S., Ma, D., Reider, R., Müller, M., & Lang, S. (2025). Generative AI for automatic simulation model generation in factory planning: A framework and prototype. *Procedia Computer Science, 22nd International Multidisciplinary Modeling & Simulation Multiconference (I3M 2025)*, *274*, 1024–1033. https://doi.org/10.1016/j.procs.2025.12.100

Lugaresi, G., & Matta, A. (2023). Automated digital twin generation of manufacturing systems with complex material flows: Graph model completion. *Computers in Industry*, *151*. Article 103977. https://doi.org/10.1016/j.compind.2023.103977

Lütjen, M., & Rippel, D. (2015). GRAMOSA framework for graphical modelling and simulation-based analysis of complex production processes. *The International Journal of Advanced Manufacturing Technology*, *81*(1–4), 171–181. https://doi.org/10.1007/s00170-015-7037-y

May, M. C., Nestroy, C., Overbeck, L., & Lanza, G. (2024). Automated model generation framework for material flow simulations of production systems. *International Journal of Production Research*, 62(1–2), 141–156. https://doi.org/10.1080/00207543.2023.2284833

Oppelt, M., Wolf, G., Drumm, O., Lutz, B., Stöß, M., & Urbas, L. (2014). Automatic model generation for virtual commissioning based on plant engineering data. *IFAC Proceedings Volumes,47*(3), 11635–11640. https://doi.org/10.3182/20140824-6-ZA-1003.01512

Pagliares, R. M., & Hirata, C. M. (2018). Mapping SPEM process specifications to activity cycle diagrams. *International Journal of Simulation Modelling*, *17*(1), 55–68. https://doi.org/10.2507/IJSIMM17(1)411

Reinhardt, H., Weber, M., & Putz, M. (2019). A survey on automatic model generation for material flow simulation in discrete manufacturing. *Procedia CIRP, 81*, 121–126. https://doi.org/10.1016/j.procir.2019.03.022

Santillán Martínez, G., Sierla, S. A., Karhela, T. A., Lappalainen, J., & Vyatkin, V. (2018). Automatic generation of a high-fidelity dynamic thermal-hydraulic process simulation model from a 3D plant model. *IEEE Access*, *6*, 45217–45232. https://doi.org/10.1109/ACCESS.2018.2865206

Tan, W. J., Seok, M. G., & Cai, W. (2023). Automatic Model Generation and Data Assimilation Framework for Cyber-Physical Production Systems. In *Proceedings of the 2023 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation* (pp. 73–83). Association for Computing Machinery. https://doi.org/10.1145/3573900.3591112