CIRP Annals - Manufacturing Technology 00 (2026) 1–5

Contents lists available at ScienceDirect

CIRP Annals - Manufacturing Technology

journal homepage: https://www.editorialmanager.com/CIRP/default.aspx

Multi-agentic production planning utilising simulation and optimisation

Merlin Korth, Martin Benfer*, Gisela Lanza (1)

wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Artificial intelligence
Production planning
Scheduling
Large language models

ABSTRACT

While applications of artificial intelligence proliferate, planning complex, volatile production remains a challenge. Hybrid methods that enable large language models to interact with classic models, such as simulation and optimisation, could address this problem. This work develops a workflow-based multi-agent architecture that generates production plans from unstructured user prompts. The system is tested using data from an automotive supplier, with 7 distinct experiments in lot sizing and setup sequencing, using simulated validation. The experiments show the system can reliably create suitable production plans while reducing execution times compared to manual planning. This work presents a step towards hybrid artificial intelligence.

1. Introduction

Increasing product variety and market volatility create highly complex tasks in production planning and control (PPC). While planners employ artificial intelligence (AI), simulation, optimisation, and metaheuristics for lot sizing, sequencing, and scheduling, these approaches require extensive domain knowledge and methodological expertise. Furthermore, planners often lack the time to respond to changing conditions or to plan for future scenarios proactively [1].

Recently, Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language processing, reasoning, and code generation, enabling the interpretation of complex instructions and contextual relationships [2]. Due to their knowledge representation structure and tendency to hallucinate, LLMs are ill-suited for planning tasks in isolation. However, their ability to bridge natural language and machine-interpretable code opens new possibilities for automating tasks that traditionally require human expertise [3], following a broader trend towards hybrid AI that pairs AI with symbolic approaches for more potent solutions [4]. In manufacturing, LLMs have already demonstrated their ability to update simulation models and create production plans [5]. Multiple specialised agents can collaborate to address the complexity of such tasks [6]. Multi-agent LLM systems (MALS) could therefore proactively plan reactions to uncertain developments by inferring data from user input, utilising tools to identify contingency plans, and validating solutions, thereby relieving experts and accelerating decision-making.

To capture this potential, this paper adopts a Design Science Research approach by iteratively designing, implementing, and evaluating the proposed MALS artefact (agent design, tool interfaces, architecture) in representative production planning scenarios to examine the hybrid system's potential. It follows two research questions: (i) How should a MALS, as an instantiation of hybrid AI for production planning tasks, be designed? (ii) Can such systems reliably conduct production planning tasks?

To address the former, the paper develops an MALS-based production planning system, equipped with corresponding classical production planning tools. To answer the second question, the system is tested through a series of scenario-planning use cases for lot sizing and setup sequencing at a large automotive manufacturer. The solution quality and speed are compared with those of expert human planners at the partner company.

The remainder of this paper is structured as follows: Section 2 discusses related work. Section 3 introduces the MALS architecture used to develop hybrid planning tools in Section 4. Section 5 shows the results of this approach in an automotive use case. Section 6 discusses the implications of this paper, and Section 7 provides a summary and outlook.

2. Related work

2.1. Discrete event simulation in production planning and control

As production systems continuously change, production planners must regularly reassess and refine system configurations, production schedules, and set-up sequences. Discrete-event simulation (DES) of material flows lets planners examine and compare potential adaptations. Combined with data capture and continuous updating, DES retains high fidelity over long time horizons, enabling reliable system representation [7]. If they are designed to respond quickly and adjust to new conditions, DESs can also generate short- and medium-term forecasts of system behaviour to inform resilient production control strategies [8].

2.2. Agentic large language models in manufacturing

LLMs and natural language processing extend conventional machine learning in manufacturing by capturing semantic relations, which benefits maintenance activities and the interpretation of PPC processes [9]. The ability of LLMs to act as autonomous agents in

* Corresponding author.
E-mail address: martin.benfer@kit.edu (M. Benfer).

https://doi.org/10.1016/j.cirp.2026.04.027

Please cite this article as: M. Korth et al., Multi-agentic production planning utilising simulation and optimisation, CIRP Annals - Manufacturing Technology (2026), https://doi.org/10.1016/j.cirp.2026.04.027

manufacturing remains limited, and explicit knowledge formalisation often dominates [10]. Recent work has applied multi-LLM agent systems to production scheduling: a multi-agent manufacturing system for intelligent shop floor management defined by four modules of agents [11], introduce a scheduling model accounting for carbon emissions utilising four agents [12] and a multi-agent LLM framework for scheduling optimisation is proposed, integrating automated problem definition, solution generation, evolutionary optimisation, and KPI-based evaluation within a feedback-driven iterative loop [13].

LLMs have also been integrated with automatic simulation model generation to enable direct creation of executable simulation models from natural language conversations, targeting accessibility barriers in production planning [14], a multi-agent RAG and Chain-of-Thought enhanced framework has been proposed to assist structural updates of advanced planning and scheduling models through natural language interaction [15], and a locally fine-tuned LLM combined with a population self-evolution mechanism has been applied to generate dynamic heuristic dispatching rules for real-time task allocation in human-robot collaborative flexible manufacturing [16]. However, none integrate LLM agents, algorithmic optimisation, and simulation-based validation within a coherent end-to-end architecture. The following section presents the proposed design of this study.

3. Multi-agent architecture design

As shown in Fig. 1, the architecture of an MALS comprises the environment, the agents' design, and their interactions with one another and with the user [17]. The LLMs have access to several tools. In the examined case, in addition to external knowledge, the LLMs use tools typically used by production planners, such as DES and MILP. DES is well-suited to represent the dynamic behaviour of systems with many elements, processes, and stepwise status changes, such as production systems. MILP models can represent typical decisions in production planning and control, enabling prescriptive planning.

Each *agent's* role within the MALS is determined by its tasks, defined in accordance with VDI 3633, and by the initial system prompts that specify how user inputs are processed. Prompt engineering techniques, including structured output formats and chain-of-thought prompting, support complex reasoning and provide necessary context, while short-term memory and the internal reason-and-act (ReAct) procedure enable agents to access external tools when required. elaborations of the necessary context to the agent. These tools, such as the DES and a PPC optimisation model, can be parameterised and queried to obtain results. Agents invoke tools via API calls to Model Context Protocol (MCP) servers, which act as the central registry for all available tools. Deterministic tools, combined with precise instructions, mitigate hallucinations [18]. Retrieval-Augmented Generation (RAG) is implemented via a vector database encoding data and external knowledge, including the production system and its DES, while enabling learning from past interactions. In addition to prompt engineering, RAG, and MCP-based tool integration, system performance could be further enhanced through domain-specific pre-training, fine-tuning, evaluation loops, and preference or reinforcement learning.

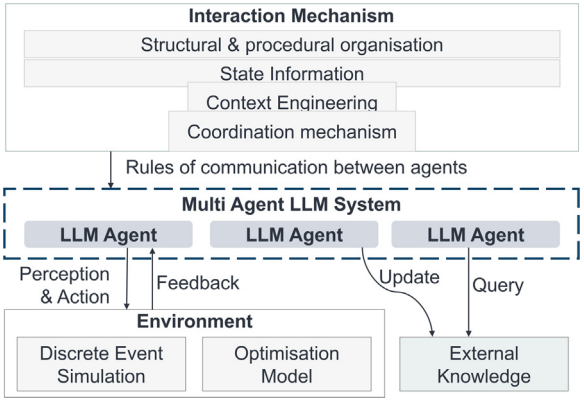

Fig. 1. Constituting elements of an MAS and its environment.

Interaction mechanisms define the structural and procedural organisation of the MALS, specifying shared state information and coordination mechanisms for communication and task assignment. A handoff tool transfers structured state information between agents, comprising reduced context, previously used tools, the main task description, and a sub-task provided by the preceding agent. Context engineering further curates and orchestrates information to ensure relevant, structured, and timely context for decision-making, enabling dynamic context adjustment and coherent operation across multi-step workflows while passing only necessary information to subsequent agents. Various *interaction architectures* are conceivable, including *Supervisor, Network,* and *Workflow*-based architectures. Based on a prior study, a workflow architecture, following the VDI 3633 procedure, is selected for its highest reliability for PPC tasks [5]. Four specialised agents sequentially process a task following a nominal action sequence: a *Strategy Agent* interprets the user task and generates the action plan, an *Analysis Agent* evaluates the production system and retrieves required information, a *Data Agent* acquires and prepares input data for the optimisation model and DES, and a *Simulation Agent* executes and interprets simulation runs. Each agent can request user support or return feedback to the preceding agent, enabling structured reasoning, targeted retrieval, and tool orchestration for reliable decision-making across multi-stage production planning workflows. [5].

4. Automated production planning studies

To evaluate the applicability and performance of the proposed approach, the following section presents the general workflow, as visualised in Fig. 2. The MALS uses Gemini 2.5 Flash for all agents and retains memory of that chat's interaction. Testing different models for different agents was not the scope of this study. When the users prompt the MALS, they specify the scenario and expected steps. An example prompt is provided in Fig. 2.

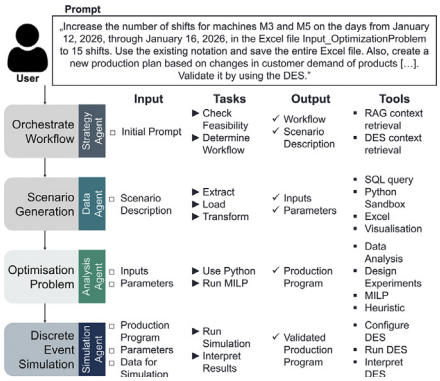

Fig. 2. Overview of the workflow and the role of the LLM Agents.

4.1. Scenario generation and translation of the LLMs

The *Strategy Agent* checks the user's scenario description, retrieves further context and provides it together with an action plan to the *Data Agent*, which is able to query SQL databases, open sandboxed XLSX or CSV files to analyse and customise input sheets, while also accessing other tools, such as Python with libraries like Pandas and NumPy to generate data that captures the scenario. The *Data Agent* then autonomously identifies the appropriate worksheets, manipulates Excel formulas to calculate required values, searches for relevant parameters, and adjusts cells and tables according to the scenario description. An excerpt of the parameters is shown in Fig. 3. The sandbox restricts access to several subsystems to prevent unintended behaviour.

Please cite this article as: M. Korth et al., Multi-agentic production planning utilising simulation and optimisation, CIRP Annals - Manufacturing Technology (2026), https://doi.org/10.1016/j.cirp.2026.04.027

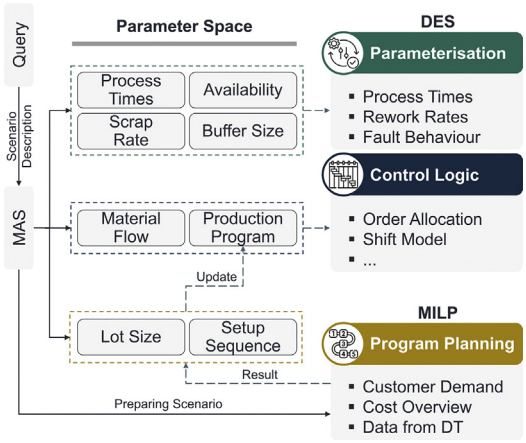

Fig. 3. Overview of an excerpt of the parameter within this approach.

4.2. Optimisation problem lot sizing and setup sequencing

The investigated tasks address the integrated lot sizing and set-up sequencing of a two-stage assembly system. The aim is to optimise production lot sizes, set-up sequences, and inventory levels simultaneously over a four-week planning horizon. In practice, planners use a mixed-integer linear programming (MILP) model to solve this complex task. Even though this model provides a prescriptive solution, planners still need to parameterise the model and evaluate the solutions, usually in combination with the DES. Thus, the MALS's *Analytics Agent* can access several heuristics or optimisation tools for different problem statements, in a manner similar to MILP, by providing the necessary input sheets and initiating the optimisation run, as shown in Fig. 2.

The MILP's objective function employs a weighted sum multi-criteria minimisation, considering (i) customer demand fulfilment, (ii) accumulated set-up costs, and (iii) storage costs. The weights are set to prioritise the former. Constraints ensure (i) products may only be produced on permissible lines, (ii) inventory is balanced for each product and component type, (iii) components are consumed according to the bill of materials, (iv) production volumes are ensured across periods, and (v) set-up times are production-sequence specific and valid. The optimisation formulation is provided in the Supplementary Material. The MILP is solved with the commercial Gurobi solver using branch-and-cut, typically to an optimality gap of 5%. Optimisation runs range from 10 to 180 min, depending on the problem's complexity, on an Azure Dav4-Series server with 16 GB of RAM.

4.3. Discrete event simulation-based validation with DES

As the MILP must make several assumptions, it is paired with a deterministic DES implemented in *Siemens Tecnomatrix Plant Simulation*. Thus, production plans can be verified in detail while considering dynamic effects. Additionally, the DES is used by planners to assess some manually proposed changes. In this DES, production lines are modelled as single stations, which capture their dynamic behaviour sufficiently well relative to the corresponding actual production lines, e.g., in terms of output and throughput time. Specifically, the effects of line internal flows are negligible for setup decisions, and the usual rolling setup changes on the production lines are adequately captured in an aggregated representation. The DES reports several KPIs for export and investigation, including output, throughput time, and utilisation rates for machines and buffers.

The *Simulation Agent* interacts with the DES via a COM interface, which exposes model objects for direct manipulation. A middleware of predefined functions abstracts these COM calls, allowing the agent to parametrise the model by passing structured inputs in JSON format. Parametrisation exploits class-based object filtering and standardised class parameters to enable systematic and replicable model configuration. The corresponding interface functions are provided in the Supplementary Material. Adjustable parameters include process times, changeover times, availability rates, MTBF, and the production program (product type, volume, sequence, and line allocation).

5. Industrial case study

The MALS is applied to the planning of a real production system and benchmarked against expert planners. The system comprises five pre-assembly lines and six final assembly lines, with technical restrictions limiting product-line assignments. >1000 weekly orders of varying product types and sizes (25–35,000 pcs) must be fulfilled from production or storage. Setup times range from 3 to 120 min, depending on product variant and line, and the planning objective is to determine suitable lot sizes and an optimal set-up sequence to minimise and avoid delivery delays. Process times are modelled as constant, aligned with empirically observed values, and a planned line availability of 87.5% is incorporated based on the industry partners' assumptions. As the DES is deterministic, a single simulation run per configuration is sufficient.

5.1. Experiments

Seven experiments that simulate real-life planning situations evaluate the MALS on tasks requiring quick, effective responses to changes in demand or production. Individual experiments comprise (A) variation of shift lengths (5S, 10S, 15S, 21S), (B) changes in product demand, (C) variation of process times, and (D) adjustment of available operating time per shift. Combined experiments include (AB) and (CD), each requiring changes to multiple tables within the same Excel file, and (ABCD), which requires multiple changes across multiple files.

The task of the MALS is to replicate the experimental steps a human planner would take, given a single prompt. An exemplary prompt and the experimental steps are shown in Fig. 3. This includes adjusting parameter sets, determining optimal lot sizes and set-up sequences using MILP, and validating the results against the DES. The general subtasks for each experimental step, along with the tools, are shown in Fig. 3.

The existing manual planning process serves as a ground truth for MALS performance assessment. Each MALS experiment is conducted 20 times to assess the system's reliability and time efficiency, and to mitigate server-induced latency effects on time efficiency. The experiments are evaluated on relative task fulfilment time (RTFT) compared to the expert planners, as well as first-time right (FTR) and second-time right (STR), which describe the success rates without and with human feedback, respectively.

5.2. Results

Expert planners performing the same tasks serve as the quantitative baseline. Planning times were recorded for each task and used to compute RTFT, enabling direct comparison between MALS and human performance. The results of the experiments are summarised in Table 1. It shows the relative time saved compared to human planners for the steps scenario generation and DES update, as well as the FTR and STR rates for those steps. The missing step, MILP

Table 1
Results relative to the expert planner baseline, reporting RTFT, FTR, and STR for scenario generation and DES update steps.

	Generate scenario		DES update	
	RTFT	*FTR/STR*	RTFT	*FTR/STR*
A	87,98%	100%	3.52%	100%
B	96,64%	100%	3.65%	100%
C	93,77%	100%	3.28%	100%
D	90,28%	100%	5.78%	100%
AB	67,11%	100%	8.23%	90%/100%
CD	90,61%	100%	7.72%	95%/100%
ABCD	75,67%	90%/100%	23.67%	80%/100%

Please cite this article as: M. Korth et al., Multi-agentic production planning utilising simulation and optimisation, CIRP Annals - Manufacturing Technology (2026), https://doi.org/10.1016/j.cirp.2026.04.027

optimisation, resulted in no time differences compared to manual use and 100% FTR.

The MALS completes all subtasks in both the scenario-generation and DES-update steps, demonstrating functional reliability. FTR values are consistently at or near 100% for individual experiments, indicating that tasks are executed correctly on the first attempt. STR values confirm that a single human feedback instance allows initially failed tasks to be completed. Here, human feedback means the user must clarify requests or accept the proposed step. It is triggered when agents detect missing information, tools, or expertise for a task. In these cases, an expert should be involved to address the issue. The RTFT in scenario generation of simple experiments ranges from 87.98 to 96.64%. Combined experiments exhibit slightly lower time savings, with the ABCD scenario achieving 75.67%, reflecting the increased complexity of combined tasks. DES updates show smaller improvements in individual experiments (3.28–5.78%), but larger RTFTs are observed in combined scenarios, reaching 23.67% in ABCD. Human planners required 16.4 s (B) to 342.9 s (ABCD) per task, while the MALS completed equivalent tasks in 0.6 s (B) to 83.7 s (ABCD).

Fig. 4 illustrates the results of one MALS-based planning. It compares the six final assembly lines before (Org.) and after hybrid optimisation and simulation (Opt.).

The number of product types remains largely unchanged, with minor adjustments to Lines 3 and 5 (top left). In the optimisation scenario, setup times for four of six lines (bottom left) are reduced, resulting in efficiency improvements of 13.8% to 58.2%. However, Line 3 shows a 540% increase, due to the increase in products manufactured on that line (top right). Machine availability, based on the simulation run, improves across all lines, due to decreased operating and set-up times (bottom right). The overall performance of the proposed architecture is assessed by combining the success rates of its constituent steps, enabling a comprehensive evaluation.

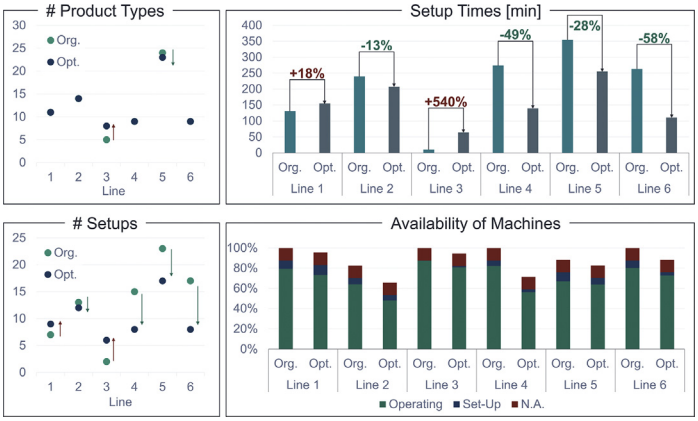

Fig. 4. Results of the optimisation and simulation runs of the final assembly line that the MALS executed compared to the baseline scenario.

6. Discussion

The proposed architecture reliably performs selected production planning tasks within the studied environment, successfully completing all sub-tasks across scenario generation and DES updates. This demonstrates that hybrid systems can assume meaningful responsibility for the tasks investigated in this domain. The MALS effectively translates between unstructured text, structured documents, and SimTalk 2.0 methods, and the achieved accuracy, combined with substantial time reductions, enables productive deployment. Nevertheless, further increases in FTR rates are required to reduce the need for human oversight and build operator trust.

The MALS achieves the highest time savings for simpler data manipulation tasks, such as (B) and (C), with reduced gains for complex combined cases like (AB). Efficiency gains were limited for data transfers between familiar systems, where program switching constitutes the primary source of delay for human planners. Here, prompt engineering and short-term memory proved critical for system performance. Prompt Engineering is critical as it determines, e. g., how information is processed (Instruction Decomposition), which output format (Output Formatting) is needed and provides boundaries (Constraint Prompting) to the agent. Without a suitable prompt engineering the agent will not achieve the results expected by the user.

Several limitations persist. The selected experiments follow a largely similar structure, and although the system could in principle, choose alternative solution paths, this flexibility is not explicitly evaluated. The approach requires well-defined, structured tools that must be created in advance. Despite the encouraging performance, the current FTR rate does not yet fully justify autonomous use in more complex scenarios without human oversight. More sophisticated interaction patterns with MALS inquiries, as well as generalisation to heterogeneous production environments and non-deterministic simulation, have not yet been tested and remain a subject of future work. Though this work has focused on a specific production system, initial results from comparable use cases indicate broader generalisability.

7. Conclusion

This paper proposes a multi-agent LLM architecture integrating algorithmic optimisation and simulation-based validation to provide automated support for selected production planning tasks, developed through a systematic design process that culminates in a workflow-based architecture for efficient and effective task fulfilment. Future work may extend the system to additional PPC processes, such as economic operating point evaluation and automated rescheduling, and provide value stream managers with integrated tools for complex planning scenarios. The current linear workflow will be extended to a diverging tree structure to accommodate user requests, new information, and agent capabilities. Additionally, the influence of different LLM models across agents and tasks will be evaluated, and the approach will be tested on further production systems to generalise the findings.

Declaration of Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Merlin Korth: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Martin Benfer:** Writing – review & editing, Writing – original draft, Validation, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Gisela Lanza:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization.

Acknowledgements

This research was funded by the German Federal Ministry for Research, Technology and Space (BMFTR) through the research project 02J23C100-114 FENI-X.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.cirp.2026.04.027.

References

[1] Kuhnle A, Kaiser J-P, Theiß F, Stricker N, Lanza G (2021) Designing an Adaptive Production Control System Using Reinforcement Learning. *Journal of Intelligent Manufacturing* 32(3):855–876. https://doi.org/10.1007/s10845-020-01612-y.
[2] Wang J, Chen Y (2023) A Review on Code Generation With LLMs: Application and Evaluation. *2023 IEEE International Conference on Medical Artificial Intelligence (MedAI)*, IEEE, Beijing, China, 284–289. https://doi.org/10.1109/MedAI59581.2023.00044.
[3] Li X, Nassehi A, Hu SJ, Joung BG, Gao RX (2025) A Large Manufacturing Decision Model for Human-Centric Decision-Making. *CIRP Annals* 74(1):621–625. https://doi.org/10.1016/j.cirp.2025.04.017.
[4] Badakhshan E, Ball P (2024) Deploying Hybrid Modelling to Support the Development of a Digital Twin for Supply Chain Master Planning Under Disruptions. *International Journal of Production Research* 62(10):3606–3637. https://doi.org/10.1080/00207543.2023.2244604.
[5] Korth M, Benfer M, Lanza G. *Designing Multi-LLM-Agent Systems for Material Flow Simulation in Production Systems," Procedia CIRP*, 137, Elsevier; Tokyo, Japan.
[6] He J, Treude C, Lo D (2025) LLM-based Multi-Agent Systems for Software Engineering: Literature Review, Vision, and the Road Ahead. *ACM Transactions on Software Engineering and Methodology* 34(5):1–30. https://doi.org/10.1145/3712003.
[7] Mourtzis D (2020) Simulation in the Design and Operation of Manufacturing Systems: State of the Art and New Trends. *International Journal of Production Research* 58(7):1927–1949. https://doi.org/10.1080/00207543.2019.1636321.
[8] Overbeck L, Graves SC, Lanza G (2024) Development and Analysis of Digital Twins of Production Systems. *International Journal of Production Research* 62(10):3544–3558. https://doi.org/10.1080/00207543.2023.2242525.
[9] Akay H, Lee SH, Kim S-G (2023) Push-Pull Digital Thread for Digital Transformation of Manufacturing Systems. *CIRP Annals* 72(1):401–404. https://doi.org/10.1016/j.cirp.2023.03.023.
[10] May MC, Neidhöfer J, Körner T, Schäfer L, Lanza G (2022) Applying Natural Language Processing in Manufacturing. *Procedia CIRP* 115:184–189. https://doi.org/10.1016/j.procir.2022.10.071.
[11] Zhao Z, Tang D, Liu C, Wang L, Zhang Z, Zhu H, Chen K, Nie Q, Ji Y (2026) A Large Language Model-Based Multi-Agent Manufacturing System for Intelligent Shopfloors. *Advanced Engineering Informatics* 69:103888. https://doi.org/10.1016/j.aei.2025.103888.
[12] Wu T, Li J, Bao J, Liu Q (2025) Large Language Model-Driven Multi-Agent Systems for Improving Production Efficiency and Reducing Carbon Emissions in Manufacturing. *Computers & Industrial Engineering* 207:111299. https://doi.org/10.1016/j.cie.2025.111299.
[13] Wang Y, Wang J, Chu Z (2025) Multi-Agent Large Language Models as Evolutionary Optimizers for Scheduling Optimization. *Computers & Industrial Engineering* 206:111197. https://doi.org/10.1016/j.cie.2025.111197.
[14] Elbasheer M, Laili Y, Longo F, Solina V, Tao Y, Veltri P, Zhang Y, Zhang L (2025) Natural Language-Driven Production Planning: Integrating Large Language Models With Automatic Simulation Model Generation in Manufacturing Systems. *Journal of Intelligent Manufacturing* . https://doi.org/10.1007/s10845-025-02732-z.
[15] Zhou Q, Li M, Li W, Qu T (2025) LLM-Assisted Advanced Planning and Scheduling Framework for Smart Manufacturing Systems. In: *Proceedings of the 2025 17th International Conference on Computer Modeling and Simulation*, ACM, Zhuhai China, 162–167. https://doi.org/10.1145/3761668.3761693.
[16] Huang J, Teng Y, Liu Q, Gao L, Li X, Zhang C, Xu G (2025) Leveraging Large Language Models for Efficient Scheduling in Human−Robot Collaborative Flexible Manufacturing Systems. *Npj Advanced Manufacturing* 2(1):47. https://doi.org/10.1038/s44334-025-00061-w.
[17] Weiss G (2013) *Multiagent Systems*, The MIT PressCambridge, Massachusetts London, England.
[18] Zhang W, Zhang J (2025) Hallucination Mitigation for Retrieval-Augmented Large Language Models: A Review. *Mathematics* 13(5):856. https://doi.org/10.3390/math13050856.

Please cite this article as: M. Korth et al., Multi-agentic production planning utilising simulation and optimisation, CIRP Annals - Manufacturing Technology (2026), https://doi.org/10.1016/j.cirp.2026.04.027