LLM-supported Usage of Simulation and Optimization in Production Planning and Control

Merlin Korth[1*], Leonard Overbeck[2], Martin Benfer[1], Gisela Lanza[1]

[1]wbk Institute of Production Science, Karlsruhe Institute of Technology, Kaiserstraße 12, 76131 Karlsruhe, Germany; *merlin.korth@kit.edu

[2]Robert Bosch GmbH, Auf der Breit 7, 76227 Karlsruhe, Germany

SNE 36(3), 2026, 139-146, DOI: 10.11128/sne.36.tn.10784
Selected ASIM SPL 2025 Postconf. Publication: 2026-02-15
Received Revised: 2026-05-27; Accepted: 2026-07-20
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Manufacturing environments are increasingly characterized by high complexity, low production volumes, and a growing diversity of product variants. Simulation is a powerful tool for analyzing such systems. However, simulations are still expert tools, offering limited accessibility for less experienced planners. This paper presents a hybrid framework that integrates numerical optimization, discrete-event simulation, and large language models to support production planners. The approach explores the solution space for optimizing workforce allocation in semi-automated work cells. A large language model-based approach is introduced to support simulation studies by reducing manual configuration efforts and enabling systematic experimentation. The method is applied in two automotive aftermarket plants with varying levels of automation to identify optimal worker-machine allocations. By integrating large language model capabilities into established simulation-optimization workflows, the approach contributes to the gradual development of autonomous simulation studies.

Introduction

A major challenge of modern production, which is characterized by high numbers of variants and an increasing individualization, is the efficient planning of the production system. Low volume and high mix production reduces economies of scale and leads to a high degree of complexity in production and logistics processes.

One of the important planning tasks is optimizing the number of workers in a work cell, as it is crucial for improving productivity and minimizing costs.

Today, the exponential increase in data availability enables production planners to make more complex and faster decisions. For this, simulation-based approaches can play a key role, but require a high level of expert knowledge and an understanding of both the simulation software and the issues faced by a production manager.

Currently, simulation studies are typically costly, time-consuming, and based on individual expertise. Experiments are performed multiple times in a repetitive, manual manner, which can reduce efficiency and tie resources.

To automate frequently recurring, similar workflows and thus release the capacities of experts for more complex simulation analysis steps, we develop an agent-based approach based on Large Language Models (LLM).

The LLM provides support in analysing the problem, reducing the solution space, creating a proposed solution, and interpreting the results. This approach is then validated at two production sites within the same international production network by integrating the actual results of simulation studies.

1 Literature Review

The foundation for this article forms [1], in which VDI defines a standardized procedure for the systematic development and application of material flow simulations. Given the focus of this contribution, the literature review will focus on the solution space exploration of the last phase *experimentation and analysis*. In addition, an overview of LLMs in production systems is provided.

1.1 Solution Space Reduction

Various methods to reduce computation time and enhance efficiency in material flow simulation studies and related domains have been explored. One proposed method is a fast-converging procedure combining genetic algorithms with material flow simulation, including early termination of unpromising runs [2]. A clustering-based method for generative design solution space reduction, using Gower distance matrix and partitioning around medoids to maintain design originality while reducing the number of solutions considered, was introduced [3].

Another methodology to assist in finding Pareto-optimal solutions in a supply chain is by reducing the design space and focussing the search of the optimization to regions of interest [4]. The development of a meta-model-based approach for generating solution spaces in sheet-bulk metal forming, enables product designers to evaluate multiple manufacturable design alternatives [5]. A semi-analytical technique that can analyse single and multi-buffer production lines, offering advantages over purely numerical approaches was introduced by [6]. Furthermore, surrogate models of discrete-event simulations, combined with multi-objective optimization methods, can provide quick approximate solutions near the Pareto-optimal front for production objectives [7].

1.2 LLM in Manufacturing

In Industry 4.0, LLMs and Natural Language Processing (NLP) have the potential to become essential to help manage knowledge [8]. The technological foundation for processing specialized vocabulary with NLP and LLMs has also already been demonstrated in manufacturing [9]. For instance, LLMs can discover causal relationships among products and subsystems in complex production processes, as demonstrated in aircraft manufacturing [10]. Additionally, their application in workforce management, such as digital shop floor management and human-robot collaboration, has shown promising results [12, 13].

A generalized approach involves fine-tuning open-domain LLMs for production applications [14]. However, in many cases their support is limited to answering questions about maintenance or quality issues and generating code for production use. In production control, there is one LLM-based approach for predicting failures and downtimes using historical data and employee annotations [13].

A related method aims to enhance failure detection and availability optimization through NLP for prediction and control [14]. Furthermore, LLMs can enhance human-centered production planning and control through reasoning integration [15]. While LLMs have been used to identify anomalies at the production system level, their integration into production planning and simulation studies remains lacking [16].

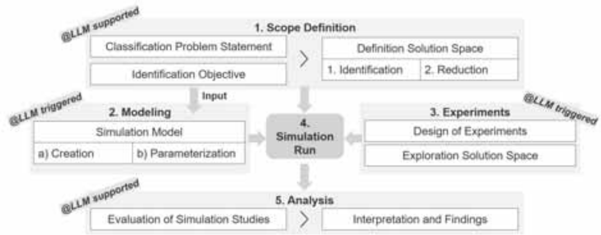

Figure 1: Overall Framework for LLM-supported conduction of simulation studies.

2 Overall Framework for Integrating LLMs in Simulations

This section presents the overall framework for integrating LLMs into material flow simulations. The framework consists of five modules, as shown in Figure 1.

The first module, *Scope Definition*, corresponds to the task definition phase in VDI3633 [1]. It receives a problem statement, an objective description as well as a description of the simulation model and the actual production system in natural language as input. By classifying the prompts, the objective and task of the simulation study can be identified. Also, relevant parameters can be derived and reduced if necessary. This reduction eliminates redundant parameter combinations as well as infeasible and dominated solutions. This way, a solution space for the experiments is obtained based on the provided information about the use case.

In the Modeling module, a simulation model is created and/or an existing model is parameterized either automatically via an import function or manually by the user. In addition, approaches such as automated simulation model generation [11] or simulation models as Digital Twins [17] can be integrated here.

Using the *Experiments* module, experimental designs are created to explore the solution space.

Depending on the complexity and size of the solution space, the design of experiments (DOE) is based on different methods, such as space-filling designs for general exploration and fractional or full factorial designs for systematic exploration.

Here, the LLM proposes a DOE to the user based on the information acquired via available tools that can be invoked via an API.

After running the experiments in the Module *Simulation Run*, the results of the simulation study, presented as KPIs (e.g., throughput, average product lead times, and machine utilization), are fed into the LLM for evaluation in module *Analysis*.

This leads to conclusions for further investigations or a final report with textual and graphic descriptions of the findings.

In the following sections, we will focus on the modules *Scope Definition*, *Experiments* and *Analysis*.

2.1 Module *Scope Definition*

In this module, to identify the solution space, the problem statement and objective are mapped within a pre-manually created (and LLM prompt-extendable) graph database to identify suitable experiments and their related variables and fixed parameters.

The outline of the mapping process involves 1st LLM-based text classification for predefined sets of problems and objectives, 2nd querying a graph database with problem and objectives classes to retrieve suitable experiments, and 3rd retrieving the set of variables and relevant KPIs.

Thereby, the solution space is identified. The outcome will be a list of proposed experiments for the user, who then decides on the experiment(s) that shall be conducted.

The text classification is based on prompt engineering techniques, such as chain of thought prompting (CoT) [18], that allow for providing and evaluating a fixed set of classes that can be identified.

The initial prompt may look like this:

> You are an AI assistant, and you are particularly good at doing text analysis. You are going to help the production manager to classify his objective and problem statements. You are only allowed to choose one of the following categories:
> objective: min cost, max output
> problem statements: buffer size, machine utilization, bottleneck, shift planning, batch size, scheduling
> Please provide your answer in Json format, for example:
> `{'value_obj': 'min cost', 'value_prob': 'num WPC', 'value_KPI': 'output'}`
>
> —–*Examples*—–
> *Example 1: I need to increase the output of my assembly line by optimizing the number of work piece carrier.*
> `{'value_obj': 'max output', 'value_prob': 'num WPC', 'value_KPI': 'output'}`
>
> —–*Content*—–
> Which are optimal worker loops and combinations of product variants in one work center consisting of 3 or 4 machines?

After identifying the objectives and problem classes they can be utilized to query suitable experiments and their variables. Here, a typical query, created automatically by the LLM, could look like:

```
query = MATCH (exp:Experiment)-[:OBJECTIVE]
        ->(ps:ProblemStatement),
        (exp)-[:VARIES]->(var:Variable),
        (ps)-[:HAS_KPI]->(kpi : KPI)
WHERE ps.name = 'scheduling'
  AND exp.objective = 'maximize output'
RETURN exp, var, kpi
```

At this point, the objective of the simulation study has been defined, the problem and relevant KPIs have been addressed, and a set of experiments and associated variables have been identified.

It is assumed that a complete description of the production system (e.g., according to the system analysis steps in VDI 3633) and a description of the material flow simulation model (e.g., process times, product variants, and fixed constraints of the solution space) are available. Next, the LLM assists in generating a first draft of an analysis of the numerical solution space and potential reductions based on further information provided by the user.

The LLM's ability to write and execute Python code allows for customized case handling, allowing for specific adjustments according to user requirements. This collaborative approach increases the effectiveness in supporting solution space reduction.

2.2 Module *Experiments*

Following the parametrization of the simulation model, a choice of Design of Experiments (DOE) for solution space exploration needs to be made. A systematic approach selects an appropriate DOE based on e.g. the similarity to previous simulation studies, data availability, and experiment characteristics.

The complexity of each experiment needs to be evaluated by including the number of factors and levels. As will be described in Section 4, we used a numerical approach for reduction of the solution space, which enables a partial factorial investigation.

2.3 Module *Analysis*

The fifth module Analysis is generating insights needed for management decisions based on simulation experiments. The LLM agent effectively supports and automates the reporting to stakeholders as well as to domain experts. Using APIs or Retrieval Augmented Generation (RAG), the LLM can incorporate the definition of KPIs and the analysis of comprehensive event logs from the simulation model to provide appropriate recommendations and empower the user with a deeper understanding of the complex mechanisms of both the simulation model and the production system.

By using tailored prompts and data-infusing APIs, the LLM is controlled in its analysis and thereby enhances its ability to interpret KPIs. This way it can provide relevant insights to guide the user in his or her own analysis.

An exemplary prompt could look like this:

```
You are an AI assistant, and you are particularly good at analyzing KPIs. You are going to help the production manager to evaluate the performance of a production system.
DESCRIPTION OF PRODUCTION SYSTEM
Here are KPIs that you need to know:
DESCRIPTION OF KPIs
Please provide your answer in Json format, for example:
'KPI': ['Interpretation': TEXT, 'Measures': TEXT]
—–Examples—–
Example 1: Utilization Rate Machine A: processing 50%, blocked 30%, waiting 20%.
'Utilization Rate Machine A': ['Interpretation': 'The machine is actively processing for 50% of the available time. It is blocked for30% of the time, indicating [...], and is waiting for [...] 20% of the time.', 'Measures': 'The measures to take are [...].']
—–Content—–
```

In addition, a benchmark of responses to sample use cases is incorporated to introduce an automated quality check before the final response is returned.

3 Case Study

In our case, the LLM (Meta Llama 3.1 70B) is given a character description and an objective for the simulation study. It uses a memory to incorporate past interactions, such as previous questions and explanations. A RAG supplements the model's internal knowledge with external data, improving accuracy, reducing hallucinations, and avoiding retraining [18].

As a prompt-engineering technique, ReAct merges reasoning traces with tool-specific actions, similarly to Chain-of-Thought prompting, to enable stepwise planning, exception handling, and external data integration [19, 20]. The LLM can invoke tools to perform task-specific actions, distinguishing this approach from established methods, such as Rapid Process Automation, which is used for routine automation. For simulation modeling, we use the software Plant Simulation by Siemens.

3.1 Module Scope Definition: The production System

Our objective is to guide users conducting simulation studies with LLMs and to combine DES with numerical optimization in a real-life setting.

The exemplary simulation study is conducted in two plants producing automotive aftermarket parts. Plant A is located in Asia, and plant B in America. To cover a maximum share of registered vehicles, a broad range of parts with each only a small number of produced parts has to be produced.

We are looking at a semi-automated process step with relatively high cycle times, which is therefore performed on multiple machines in parallel.

Modeling relevant differences between the product variants are manual times for machine parallel material preparation (PT), the time for manual unloading, cleaning and loading the machine between cycles (service time – ST) and the automated machine time (MT). While the products and production process are similar, plant A has currently a higher degree of automation than plant B, which leads to different objectives of the simulation studies.

Whereas for plant A various investment scenarios with different automation solutions and corresponding effects on the share of manual and automated work content must be evaluated, the automation scenario for plant B has already been addressed and implemented. For plant B, it must be evaluated which number of machines can be served by one operator depending on the combinations of variants on the machines (two possible solutions are depicted in Figure 2).

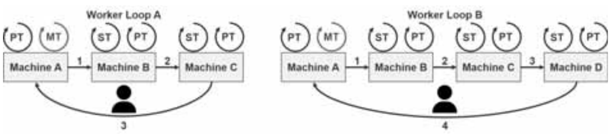

Figure 2: Different loop combinations with related time components.

This query calls automatically via API a Neo4 database instance to receive possible experiments and related variables. In our case the result provided the experiments scheduling and shift planning with the variables of product sequence and worker loop.

```
User: Which are optimal worker loops and combinations of
      product variants in one work center consisting of
      three machines once the planned automatization of raw
      material preparation is installed ?
Chatbot: {'value_obj': 'max output', 'value_prob': '
      machine utilization '}
query = MATCH (exp:Experiment)−[:OBJECTIVE]−>(ps:
      ProblemStatement), (exp)−[:VARIES]−>(var:Variable), (ps)
      −[:HAS_KPI]−>(kpi:KPI)
WHERE ps.name = 'machine utilization'
  AND exp.objective = 'max output'
RETURN exp, var, kpi
```

3.2 Module Experiments: Analytical solution space reduction

While in both plants simulation is used to determine machine and operator utilization (time share active in proportion to total time) as well as machine and operator productivity (parts per machine/operator hour) in different configurations, a python script is used to calculate the expected waiting times of machines and operators beforehand (as shown in Eq.1 to Eq.5).

An LLM agent will, based on the already acquired information from the previous steps, support the development of a Python script by generating code for the calculation of expected waiting times.

Furthermore, it can execute the script via an API to process input data, enabling an iterative development process through interactive feedback from experts and supporting tools. Through the definition of product families (similar variants grouped based on identical ST, PT, and MT), a complete exploration of the (reduced) solution space is possible.

This allows us to rank all practical solutions (combinations of product families on the machines) according to one relevant KPI (total waiting times). As this is a static calculation and the comparison is based on only one KPI, a further simulation-based investigation is still needed, but can be performed faster as it can start with promising candidates and ignore dominated ones.

$$\min X_A + X_B + X_C + X_D \tag{1}$$

With respect to:

$$X_A = |MT_A - (PT_B + PT_C + PT_D + ST_B + ST_C + ST_D)| \tag{2}$$

$$X_B = |MT_B - (PT_A + PT_C + PT_D + ST_A + ST_C + ST_D)| \tag{3}$$

$$X_C = |MT_C - (PT_A + PT_B + PT_D + ST_A + ST_B + ST_D)| \tag{4}$$

$$X_D = |MT_D - (PT_A + PT_B + PT_C + ST_A + ST_B + ST_C)| \tag{5}$$

3.3 Module Analysis: Simulating and generating insights

The configurations that provide the best results in the numerical evaluation will be evaluated in simulation with different random seeds and production plans to verify waiting times and obtain further decision-relevant KPIs (e.g., OEE, utilization rate, throughput times, WIP). The consecutive simulation runs are therefore an indispensable, valuable augmentation of the prior numeric calculations. Trade-offs between different KPIs and levels of improvements must be made, for example, between the utilization rate of operators and of machines. An optimal balancing to eliminate waiting times completely is not always possible due to process requirements and customer delivery dates

Figure 3 shows exemplary how the simulation results can be used for a multicriteria decision that has to weight multiple relevant KPIs. In this two-dimensional graphic, only two KPIs can be compared. While this is a typical way to represent the results of simulation studies used by simulation experts, it is an important starting point for a far more indepth representation and discussion of inevitable trade-offs between objectives empowered by LLMs.

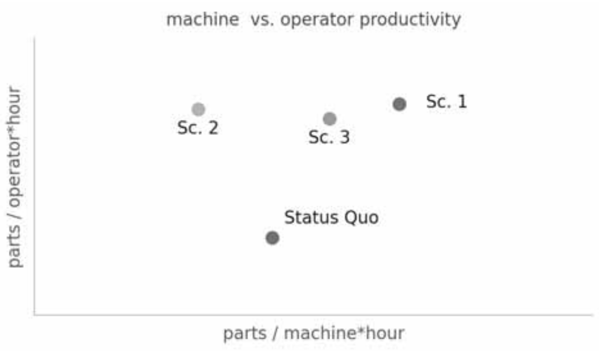

Figure 3: Evaluation of different automation and workflow scenarios.

4 Results

4.1 Module Experiments: Analytical solution space reduction

Using a prior numerical solution space exploration guided by only one selected KPI and executed in a simplified solution space, the number of system configurations that shall be simulated can be drastically reduced. With our approach, the number of simulation experiments could be reduced for plant A from 2,685,617,610 for all possible combinations, to 85,140 for all combinations of families, and finally to 52 promising combinations to simulate.

4.2 Module Analysis: Results of the experiments

After promising combinations of variants with corresponding layouts and work standards were identified using the numerical calculations, they were simulated to obtain further KPIs and hence enable a holistic evaluation and comparison of all options. Exemplary results for one scenario are shown in Table 1.

Scenario		OEE [%]	#Parts	Util. [%]
XY	Machine A	85.66	5,800	86%
	Machine B	68.20	5,800	69%
	Machine C	96.40	13,920	96%
Operator				75%
Average / Sum		83.42	25,520	82%

Table 1: KPI results for scenario XY showing OEE, total parts produced, and utilization per machine and operator.

5 Discussion and Conclusion

The paper presents a framework for LLM-supported simulation studies. This approach enables automated and user-guided experiment design, improving simulation efficiency and providing valuable insights for further investigation and reporting.

The case study showed the framework's potential and highlights further practical challenges of using LLMs in simulation studies. Key challenges faced are the need for quality assurance, careful design of prompts, and the difficulty of accessing data or providing sufficient context.

Additionally, creating generic, reusable simulation experiments on their own remains a challenge, as LLMs still struggle with zero-shot prompting in very complex use cases. While the LLM shows promising results and has demonstrated clear improvements in production systems through the studies, they are not yet fully capable of autonomously generating and defining simulation experiments.

Future work aims to extend the proposed framework by developing a multi-agent LLM architecture that allows more complex investigations. This includes e.g., the automation of a data pipeline and creating formalized descriptions of simulation studies to document the simulation experiments.

Acknowledgement

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number LA 2351/88-1.

Publication Remark

This contribution is the revised extended version of the conference version for **ASIM SPL 2025** - 21.ASIM-Fachtagung Simulation in Produktion und Logistik - Dresden 2025, published in *Simulation in Produktion und Logistik 2025*, ASIM-Mitteilung Nr.194,
ISBN 978-3-86780-806-4, e-ISBN 978-3-86780-809-5,
Volume DOI `10.25368/2025.233`,
Article DOI `10.25368/2025.256`

References

[1] Verein Deutscher Ingenieure e. V. *VDI-Richtlinie 3633: Simulation von Logistik-, Materialfluss- und Produktionssystemen, Blatt 1*. Düsseldorf: VDI-Verlag; 2014.

[2] Frank M, Laroque C, Uhlig T. Reducing computation time in simulation-based optimization of manufacturing systems. In: *Proceedings of the 2013 Winter Simulation Conference (WSC)*; 2013 Dec 8–11; Washington, DC. Piscataway: IEEE. p. 2710–2721. doi: 10.1109/WSC.2013.6721660.

[3] Botyarov M, Miller EE. Partitioning around medoids as a systematic approach to generative design solution space reduction. *Results in Engineering*. 2022; 16: 100681. doi: 10.1016/j.rineng.2022.100681.

[4] Aslam T, Ng AH. Combining system dynamics and multi-objective optimization with design space reduction. *Ind. Manag. Data Syst.* 2016; 116(2): 291–321. doi: 10.1108/IMDS-12-2014-0375.

[5] Sauer C, Schleich BR, Wartzack S. Meta-model-based generation of solution spaces in sheet-bulk metal forming. *Procedia CIRP*. 2020; 91: 905–910.

[6] Ložar V, Abdulaj F, Opetuk T, Hadžić N, Cajner H. Application of semi-analytical methods in production systems engineering: Serial lines. *Tehnički glasnik*. 2020; 14(3): 360–364. doi: 10.31803/tg-20200720151140.

[7] Červeňanská Z, Kotianová J, Važan P, Juhásová B, Juhás M. Multi-objective optimization of production objectives based on surrogate model. *Applied Sciences*. 2020; 10(21): 7870. doi: 10.3390/app10217870.

[8] Ansari F. Wissensmanagement in der Industrie 4.0. In: *Handbuch Unternehmensorganisation: Strategien, Planung, Umsetzung*. Berlin, Heidelberg: Springer Vieweg; 2024. p. 1–15. doi: 10.1007/978-3-642-45370-0_97-1.

[9] Brundage MP, Sexton T, Hodkiewicz M, Dima A, Lukens S. Technical language processing: Unlocking maintenance knowledge. *Manufacturing Letters*. 2021; 27: 42–46. doi: 10.1016/j.mfglet.2020.11.001.

[10] Zhou B, Li X, Liu T, Xu K, Liu W, Bao J. CausalKGPT: Industrial structure causal knowledge-enhanced large language model for cause analysis of quality problems in aerospace product manufacturing. *Advanced Engineering Informatics*. 2024; 59: 102333. doi: 10.1016/j.aei.2023.102333.

[11] May MC, Nestroy C, Overbeck L, Lanza G. Automated model generation framework for material flow simulations of production systems. *Int. J. Prod. Res.* 2023; 62(1–2): 141–156. doi: 10.1080/00207543.2023.2284833.

[12] Xia L, Li C, Zhang C, Liu S, Zheng P. Leveraging error-assisted fine-tuning large language models for manufacturing excellence. *Robot. Comput.-Integr. Manuf.* 2024; 88: 102728. doi: 10.1016/j.rcim.2024.102728.

[13] May MC, Neidhöfer J, Körner T, Schäfer L, Lanza G. Applying natural language processing in manufacturing. *Procedia CIRP*. 2022; 115: 184–189. doi: 10.1016/j.procir.2022.10.071.

[14] Ansari F, Kohl L, Giner J, Meier H. Text mining for AI enhanced failure detection and availability optimization in production systems. *CIRP Annals*. 2021; 70(1): 373–376. doi: 10.1016/j.cirp.2021.04.045.

[15] Kohl L, Eschenbacher S, Besinger P, Ansari F. Large Language Model-based Chatbot for Improving Human-Centricity in Maintenance Planning and Operations. In: *Proceedings of the European Conference of the PHM Society 2024*; 2024 Jun; Prague, Czech Republic. Vol. 8, No. 1. p. 12. doi: 10.36001/phme.2024.v8i1.4098.

[16] Rane N, Choudhary S, Rane J. Intelligent Manufacturing through Generative Artificial Intelligence, Such as ChatGPT or Bard. SSRN Preprint. 2024. doi: 10.2139/ssrn.4681747.

[17] Overbeck L, Graves SC, Lanza G. Development and analysis of digital twins of production systems. *Int. J. Prod. Res.* 2024; 62(10): 3544–3558. doi: 10.1080/00207543.2023.2242525.

[18] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, Küttler H, Lewis M, Yih W-t, Rocktäschel T, Riedel S, Kiela D. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*; 2020. Art. 793. Curran Associates Inc. doi: 10.5555/3495724.3496517.

[19] Wei J, Wang X, Schuurmans D, Bosma M, Ichter B, Xia F, Chi EH, Le QV, Zhou D. Chain-of-thought prompting elicits reasoning in large language models. In: *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*; 2022. Curran Associates Inc. doi: 10.5555/3600270.3602070.

[20] Yao S, Zhao J, Yu D, Du N, Shafran I, Narasimhan K, Cao Y. ReAct: Synergizing reasoning and acting in language models. In: *International Conference on Learning Representations (ICLR)*; 2023. url: https://arxiv.org/abs/2210.03629.