

CIRP Conference on Manufacturing Systems (CIRP CMS)

Industrial Voice User Interface for Programmable Machine Tools: Application to a 3-Axis Computer Numeric Code Milling Machine

Avik Mukherjee^{a*}, Richard Vlas^b, Polina Häfner^c, Andreas Kimmig^d, Marcel Wagner^a, Jan. C. Aurich^a

^aInstitute for Manufacturing Technology and Production Systems (FBK), University of Kaiserslautern-Landau, P.O. Box 3049, 67663 Kaiserslautern, Germany

^bSABO Mobile IT GmbH, Engelstr. 6, 77815 Buhl, Germany

^cKarlsruhe Institute of Technology, Kriegsstraße 77, D-76133 Karlsruhe, Germany

^dagitur GmbH, Schönbornstraße 11, 76661 Philippsburg, Germany

* Corresponding author. Tel.: +49 631 205 3224; fax: +49 631 205 3304. E-mail address: avik.mukherjee@rptu.de

Abstract

The human machine interface (HMI) for interacting with computer numeric code (CNC) machines is predominantly constrained to touch-based user interfaces. These interfaces are limited in terms of operator mobility, -flexibility, -inclusivity, and -ease of interaction. Voice user interfaces (VUIs) have proven to be advantageous by mitigating the use of hands, the need for a navigation-based instruction manual, and the stationarity of the operator. Agentic artificial intelligence (Agentic AI) can be used for developing a flexible, adaptive, and inclusive VUI for speaking with CNC machines. However, implementation of such an approach bears challenges like the predictive accuracy of the large language models (LLMs), validation, and deployment of the software. Iterative training of LLMs, and validation of the solution is done by using a software in loop pipeline facilitated by the virtual twin of the CNC machine. This article discusses in detail the conceptual framework, software architecture, and implementation details of a VUI for CNC-enabled 3-axis milling machine. It also discusses the challenges faced with training of the LLMs and validation of the software. Additionally, this article discusses the shortcomings that need to be addressed to make such a solution deployable in an industrial scenario.

© 2026 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 2025 CIRP CMS

Keywords: Human-Machine Interaction; Large Language Models; Agentic AI; Voice User Interface; Industry 4.0

1. Introduction

The recent advancements in information and communication technology, the internet of things (IoT), and artificial intelligence (AI) have led to the creation of cyber-physical machine tools (CPMT). CPMTs focus on integrating digital twins, real-time data analytics, AI-driven adaptive control, and secure IoT connectivity to enhance precision, resilience, and autonomy in machining operations. Liu et. al. [1] propose a CPMT architecture consisting of five central components namely computer numeric control (CNC) enabled machine tool, data acquisition devices like sensors, cameras,

machine tool digital twin for various analytical, predictive, real-time controlling tasks, and smart human machine interfaces (HMIs). CNC is the automated operation of machine tools using computer-generated instructions to perform precise and repeatable machining tasks. CNC machines primarily operate using G-code for motion and path control, M-code for auxiliary functions such as spindle control or coolant activation, and occasionally custom macros or parametric codes for complex, reusable routines [2].

As the real-time computational capability and services provided by CNC machines increase, a demand arises for developing smart HMIs that enable user-friendliness while

ensuring security in safety-critical machining operations [1]. HMIs have subsequently evolved from simple button-based panels to advanced touch-based graphical user interfaces that integrate with digital twin of CNC machines to provide real time process monitoring and diagnostic information [3].

Voice-based interactions have been successfully used in the field of consumer electronics. VUIs can prove beneficial for industrial machine as well, as they significantly reduce interaction time and operator's mental workload [4]. VUIs also facilitate inclusivity and flexibility of operations. However, the present research regarding VUIs is limited and does not take into account noisy shop floor environments, low prediction accuracy of the underlying speech recognition algorithms, and security and safety concerns [4]. Large language models (LLMs) provide near-human accuracy in natural language processing tasks like natural language understanding and intent recognition. A VUI developed using agentic AI architecture that considers LLMs as the main conversational agent can address issues of prediction accuracy of the underlying conversation engine. It can also provide added security from users with malicious intents. Additionally, it can help facilitate natural language dialogues between operator and machines. However, one of the challenges of such an approach is achieving a connection between speech-based commands—representing the user's intended machine actions—and the CNC control logic so that operations execute accurately and safely. Additionally, challenges like hallucinations, constrained deployment scenarios, and fine-tuning of LLMs need to be addressed when developing such a solution for industrial machines. This research tries to address the research gap regarding the development of VUIs for industrial CNC machines by using LLMs as the core conversational component. Towards achieving that, this article discusses the conceptual pipeline, implementation details, application challenges, and suggestions for practical deployment of LLM-based VUI solutions for CNC machines.

2. State of the Art

2.1 Human Machine Interfaces for CNC Machines

Research regarding HMIs for CNC machines is currently characterized by the integration of advanced graphical user interfaces (GUIs), multimodal interaction, and digital twin connectivity. Advanced GUIs in CNC machines provide touch-enabled, adaptive dashboards and 3D visualizations [5]. Multimodal interactions integrate voice, gesture, and augmented reality/mixed reality (AR/MR) overlays for more natural control [6]. Additionally, digital twin connectivity links the HMI with real-time simulation and predictive feedback to enhance operator decision-making [7]. Machine learning (ML) plays a critical role in enabling these advancements. Touch-enabled adaptive dashboards benefit from ML models that filter sensor data, highlight anomalies, and customize information according to user profiles [5]. AR/MR overlays use computer vision to accurately align digital information with physical machine context [6]. The integration of HMIs with digital twins increasingly relies on predictive ML models, which simulate machining outcomes, anticipate tool wear, and

recommend corrective actions, thereby turning the interface into a proactive decision-support system [7]. Despite the large-scale impact of LLMs in manufacturing, a gap is noted in the literature regarding the use of LLMs in HMIs for CNC machine tools.

2.3 LLM Application in CNC Machines

The enhanced natural language processing abilities of LLMs have facilitated their use in various applications related to CNC machine tools. Abdelaal et. al. [8] demonstrates a pipeline for self-corrective G-code generation from free-flowing natural language texts. LLMs are used for integrating robot control code into the G-code framework as illustrated by [9]. Beyond code generation, LLMs also contribute to intelligent decision support. LLMs integrated with domain-specific knowledge graphs enable contextualized process planning by recommending cutting parameters, tools, and operations based on part geometry and machine constraints [10]. In HMI contexts, LLMs are being explored as conversational agents that enhance operator support systems, offering explainable feedback, predictive maintenance suggestions, and dynamic task guidance [11]. LLMs, however need to be iteratively trained and validated to improve their prediction accuracy. Due to the costs incurred with the use of the physical machines, virtual commissioning (VC) can be thought of as an effective alternative for this iterative training and validation.

2.2 Virtual Commissioning for CNC Machines

VC is the process of using a digital model or digital twin of a machine or system to simulate, test, and validate control logic, mechanical behaviour, and system interactions before physical deployment. VC can be mainly broken down into three categories: hardware-in-loop (HIL), software-in-loop (SIL), and model-in-loop [12]. Digital twins enable dynamic synchronization between virtual models and their physical counterparts, thereby overcoming the model inconsistencies commonly encountered in conventional VC approaches and significantly reducing the time and effort required during the commissioning process [13]. Multiple use cases of digital twin-enabled VC for CNC machine tools can be found in literature. As an example, [14] uses HIL to commission the performance of a controller. Additionally, Striffler et. al. [12] states that SIL commissioning of control logic allows validation of the control system in parallel to the manufacturing and assembly of a specific production system. Digital twin-based VC can also be used for commissioning a CNC machine for iterative fine training and validation of LLM based interfaces that communicate between the operator and the machine.

2.4 VUIs as Voice Assistants

An approach for developing VUI for CNC machines is provided in [4], which develops an integrated interface for issuing voice commands to a CNC milling machine and a robot. This approach focuses on a fixed set of commands pertaining to three navigational layers of a conventional HMI. As an advantage, it notes the increased flexibility of use and faster

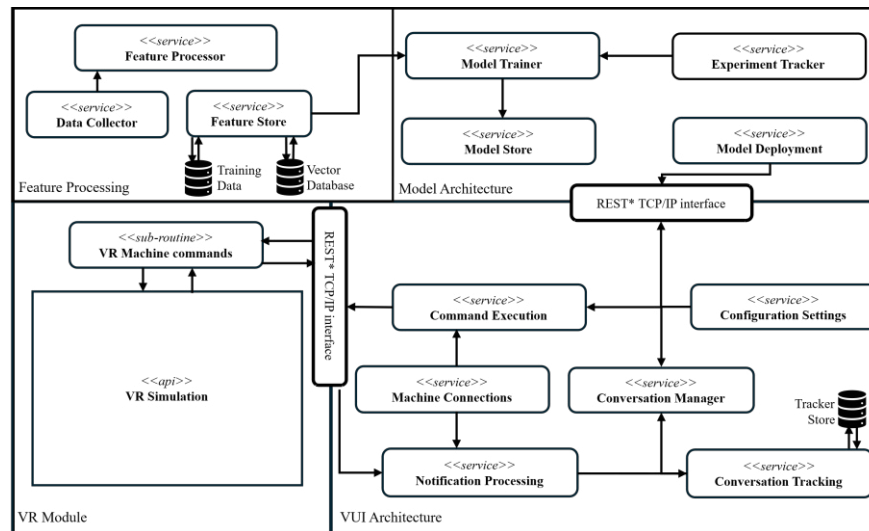


Fig. 1. Software architecture of VUI

communication between the operators. A drawback is also found in the prediction accuracy of such an approach due to the use of voice recognition systems. Another such approach found in the literature focuses on implementing a simple prototypic voice assistant for a CNC milling machine that supports operators and maintenance personnel mainly by assisting in alarm situations [15]. This study also creates a framework for measuring the quality of any voice assistant that can be implemented for an industrial machine. Other relevant studies can be found in [16] for using VUI in production logistics, and [17] which reports on a conventional VUI as a digital assistant to a digital twin and notices the barriers and shortcomings in the approach. The current literature predominantly points out a gap in the study of VUIs for interacting with CNC machines [4]. The software described in this article addresses this gap by using LLM as the core conversational engine for the VUI, which is further validated using a SIL approach using an immersive virtual twin (VT).

In summary, LLMs can be used for developing efficient adaptive voice user interfaces that can facilitate natural language interactions between operator and machines. However, LLMs need to be trained on machine-specific vocabulary and validated to be able to be successfully deployed in an industrial scenario. Iterative training and validation can prove to be cost- and time- intensive when a physical machine is involved due to restricted availabilities and safety concerns when testing on the machine. Such costs can be eradicated by using VC of machines for iterative training and validation and using the physical machine only for final validation and deployment.

3. Software Solution

This section describes the three most important conceptual components of the VUI solution: software architecture, conversation engine, and virtual commissioning. Afterwards, a brief overview of the overall implementation pipeline is provided that helps the reader understand the flow of the data and functions in the solution.

3.1. Software Architecture

The software architecture as illustrated in Fig. 1 can be divided into four modules. VUI architecture represents the container for all sub-services that are necessary for converting the speech data into a format suitable for communicating with the LLM and the VC components. The *Conversation Manager* service is the main component of this module. It acts as the central communication point with the deployed LLM and the VT of the software. It formats the prompt to be sent to the LLM depending on the user query and the machine state. The *Configuration Settings* service provides the configuration parameters and tolerances specific to the use case that the VUI is connected to. This in turn facilitates the generalizability and the transferability of the solution. The *Conversation Tracking* service uses a tracker database (DB) for storing conversations to preserve the context of the dialogue. The *Command Execution* service processes the output of the LLMs and converts them to machine-specific commands. The LLMs are hosted and trained in the *Feature Processor* and *Model Architecture* services. Pretrained LLMs are fine-trained on domain-specific vocabulary of the CNC milling machine using prompt engineering. The interactions between the user and the LLMs are collected using the *Data Collector* service and tokenized in the *Feature Processor* and stored using the *Feature Store* service. The *Data Collector* service also collects data from various sources relevant to the CNC milling machine like assembly instructions and specifications of the CNC milling machine and the Linux CNC user manual. These sources are similarly tokenized and then embedded into vectors. These vectors are stored in the vector database that is then used for the retrieval augmented generation (RAG) component of the LLM engine. The data from the *Feature Processor* service is also utilized for iterative fine training of the model. A list of pre-trained LLMs for further fine training in the *Model Trainer* are stored and managed using the *Model Store*. The *Experiment Tracker* logs the model parameters and results of experiments for reproducibility. The pre-trained model is selected as an internal parameter from the

Configuration Settings service in the VUI-architecture. VC block mainly comprises two sub-modules. The VR machine commands module takes input from the VUI architecture and converts it to information specific to virtual twin kinematics. This module acts as the main SIL component of the solution.

3.2. Conversation Engine

The conversation manager service communicates with the LLM using the Agentic AI framework of Langgraph [18]. The graph-based interaction with the LLMs is facilitated by the development of use-case-specific business flows that imitate

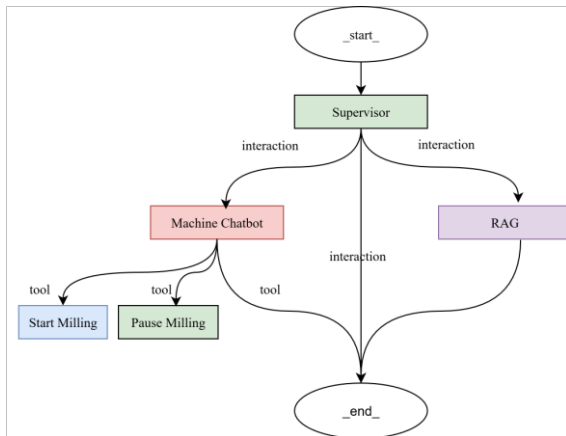


Fig. 2. Langgraph architecture for CNC chat bot

the conversation between the operator and the assistant. The possible interactions with the LLMs are identified and used to build a graph architecture. The lang-graph starts with a supervisor node that redirects the flow to the appropriate subgraph. The lang-graph architecture is illustrated in Fig. 2. The deployed LLMs are functionally divided into machine chatbot and RAG subgraphs. RAG uses the vector DB in the *Feature Store* service (Fig. 1.) to retrieve relevant information as per the query of the user. The machine chatbot sub-graph is further divided into tools that correspond to commands that the operator wants to relay to the machine.

3.3. Virtual Commissioning

The VC is used for the validation of the solution. The Linux CNC controller facilitates both the SIL and the HIL validation of the solution. In the SIL phase the Linux CNC is used as a

virtual controller that can accurately represent the underlying physics of the CNC machine. The Cartesian kinematics of the controller are then transferred into the virtual domain of the virtual twin (VT) for the machine. The VT of the machine is built using the PolyVR virtual engineering framework, which



Fig. 3. Virtual twin of 3-Axis CNC machine

is an open-source tool for creating machine simulations [19]. The VT mirrors all functionalities available through the machine's control interface, which can be linked via standard communication protocols such as MQTT for synchronized operation and data exchange. Additionally, it simulates the material degradation effect on the workpiece to simulate the milling operation. Fig. 3 shows the virtual twin built for the CNC machine.

3.4. Implementation Example

The conceptual flow of the software is illustrated in Fig. 4. The voice command issued by the user is first transcribed into text using the state-of-the-art speech-to-text and text-to-speech provider MS Azure. A gap of 10 seconds is observed to mark the end of recording for a given statement. The recorded statement is then transcribed to text and sent to the *Conversation Manager* service (Fig. 4). The *Conversation Manager* stores the transcribed text in the *Conversation Store* to retain the context of the conversations. Additionally, it creates the formatted prompt depending on the user query and invokes the lang-graph architecture to enable communication

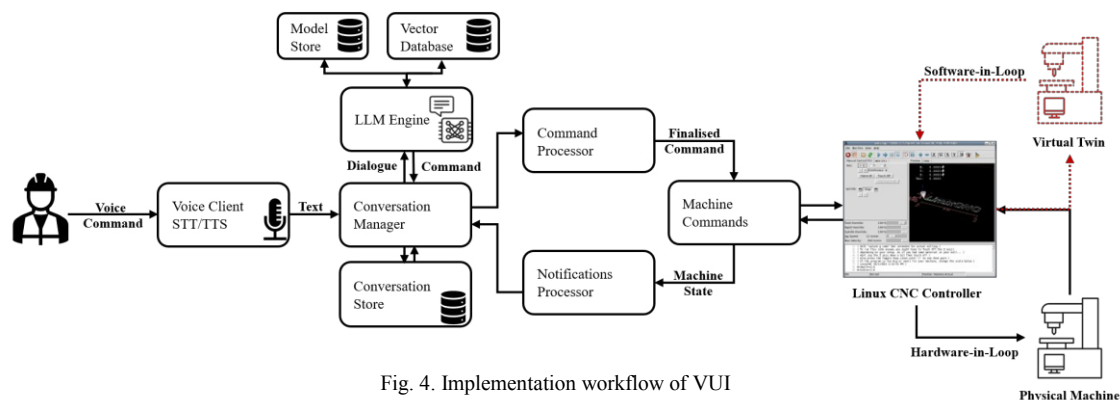


Fig. 4. Implementation workflow of VUI

with the LLMs. The LLM engine accesses the configured LLM from the *Model Store* and, depending on the subgraph being executed, refers to the vector DB for additional general information on the CNC machine or redirects to the tool subgraph (Fig. 2) for execution. This facilitates a bi-faceted application of the VUI as both a personalized voice assistant fitting for the CNC machine and as a controller for speaking commands to the machine. The response received from the LLM engines is then parsed into a JSON format suitable for communicating with the other services in the *Command Processor* service. This structure is then translated into machine commands and sent to the Linux CNC controller. In the SIL setup of the software, the Linux CNC controller further communicates with the VT of the solution, thus providing an immersive virtual environment for iterative improvement and validation of the solution. The SIL-validated solution is then transferred to the Linux CNC controller of the physical machine for the final fine tuning of the solution. This approach facilitates easier and cheaper fine-tuning iterations for the LLMs and reduces the risk of unsafe operations during the intermediate validation phase of the solution. Additionally, the Linux CNC controller provides the machine state information to the conversation manager service that is critical in deciding the execution flow of commands. One of the challenges that needs to be overcome in this approach is the accuracy of pre-trained LLMs. There can be possible hallucinations even after placing safeguards on the LLM response that can endanger the machine and the operator. Other important factors that need to be tested for the LLMs are included but not limited to the crispness of the response, the adherence to the language in case of multilingual speech, the comprehensiveness of the response. A validation pipeline is built to facilitate the above required measures. The validation experiments are detailed in the next section.

4. Experiments

The probabilistic output provided by LLMs necessitates the use of proper steps for ensuring the accuracy of the response output. A command misinterpretation of a lateral movement of the rotating milling tool by 5 mm instead of 2 mm can damage the tool and the workpiece subjected to the milling operation. Additionally, the brevity of the response of the assistants plays a major role in deciding the latency of the VUI. High latency can lead to loss of valuable machine time which can prove harmful for achieving the takt in an industrial shop floor. To mitigate these risks, the models need to be validated using a specific validation pipeline. Phoenix Arize-AI [20] benchmarking tool is used for the validation of LLMs and the lang-graph. The validation pipeline is also used for prompt

engineering of the LLMs. Fig. 5 shows the validation pipeline used for the iterative validation of the LLMs.

The validation methods used are quantitative validation and LLM as a judge technique. A set of 20 commands with their corresponding responses are noted, and they are validated for tasks like hallucination detection, comprehensiveness, language adherence, tool calling, and conciseness. The hallucination evaluation ensures that the LLMs always provide correct information. Consequently, a higher evaluation score in Table 1 ensures a model that hallucinates less as compared to a model with a lower score. Comprehensiveness measures if the text responses from the LLMs provide the necessary information. Language adherence ensures that the response adheres to the language of the question. Tool-calling ensures that the LLM can identify the right command for execution. The conciseness evaluator is used to evaluate the conciseness of the responses. A set of 8 LLMs are evaluated for the datasets and for prompt generation. The results are noted in Table 1. GPT-4 provides the best results for the evaluation. It is considered as the reference LLM for the LLM as a judge evaluation. The best tool-calling support is, however, provided by Llama3.2 It provides a 100% accuracy on the evaluation dataset (Table 1) However, Llama 3.2 suffers from the problem of hallucination, and wordiness of the responses (Table 1).

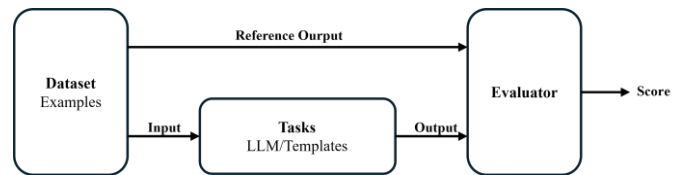


Fig. 5. Validation pipeline for LLMs

Qwen2.5:14B performs the best on the datasets out of the smaller open-source models as can be seen from Table 1. As per the functionality of the LLMs in the VUI, tool calling, hallucination, and comprehensiveness have a higher weight among the evaluation metrics as compared to the rest.

5. Discussion

The implementation of the VUI solution for the CNC machine poses certain application challenges. A small list of such challenges and improvements is recorded in Table 2. The accuracy of the LLMs poses a critical challenge that should be addressed to enable a deployable VUI solution. As the results in Table 2 show, proprietary LLMs like GPT-4.0 trained over trillion parameters have the best evaluation results. However, storing and managing such a model is cost intensive. The results considerably decline when considering open-source

Table 1: Evaluation of different LLMs on CNC benchmark data

Model	Hallucination	Comprehensiveness	Language Adherence	Tool Calling	Conciseness
Llama 3.2	40%	82%	100%	100%	40%
Qwen2.5	43%	100%	100%	86%	70%
Qwen2.5:14B	71%	91%	100%	86%	70%
Mistral:	57%	82%	100%	86%	80
Deepseek-r1	Not Supported	91%	100%	Not Supported	80%
Llava:13B	Not Supported	82%	100%	Not Supported	80%
GPT4	89%	98%	100%	97%	93%
Alibrayam	Not Supported	18%	100%	Not Supported	20%

models of lower sizes. Therefore, a tradeoff needs to be considered between the size of the model and the accuracy of its responses. Smaller models tend to hallucinate and can often interpret wrong commands and provide false information to the user. Specific techniques like reinforcement learning based on human feedback [21] can help in mitigating the issue of hallucination by aligning the model's output with human evaluation, especially around factual accuracy.

Table 2: Summary of shortcomings and improvements

Category	Challenge	Improvement
Validation	Comprehensive validation scenarios	ML driven agent simulation
Model quality	Hallucination	Methods to mitigate hallucinations
Model choice	Tradeoff between size and accuracy	Cloud independent Edge VUIs
Infrastructure	Data protection and latency issues in cloud solutions	Cloud independent deployment
Standardization	Limited transferability and reuse	AAS Type-3 Integration (IDTA/RAMI 4.0)

The current approach uses a cloud-deployed solution that leads to issues with latency when executing real-time commands. One way to overcome this challenge is by using an offline deployable solution that can be hosted on edge computers connected to the machine. Additionally, manual validations limit the voice-based interactions between the user and the VUI. One way to mitigate such limited validation is to use reinforcement learning-driven agent simulations. Standardization poses another critical application challenge in the current approach. The software components do not rely on standardized interfaces or semantics that lead to inconsistent semantics and poor reuse, thus proving expensive when considered as a deployable solution for an industrial shop floor. This can be mitigated by using standardized interfaces called Asset administration shells (AAS).

6. Conclusion and Outlook

It can be noted that a LLM-driven VUI can be used as a flexible, inclusive alternative to the current HMIs for machine tool interface. High comprehensiveness, function-calling, and low hallucination capabilities guaranteed by pre-trained LLMs, facilitate their use in an agent-based lang-graph framework as the conversational engine. The need for iterative fine training, and validation is resolved using a SIL pipeline facilitated by the VT of the CNC machine. However, the migration of such a solution to industrial use cases poses certain challenges. These challenges, stated in Section 5, should be mitigated for the deployment of such a solution in industrial scenarios.

Acknowledgements

This research is funded by the German Federal Ministry for Education and Research (Bundesministerium für Bildung und Forschung, BMBF) within the project KMU-Innovativ - Verbundprojekt RelSIM: „Effiziente Entwicklung, Validierung und Einführung von sprachgesteuerten Benutzeroberflächen – Ein Reinforcement Learning basierter Ansatz mittels Simulation menschlicher Agenten“ (Grant number 01IS25022D).

References

- [1] Liu, C., Xu, X., 2017. Cyber-physical Machine Tool – The Era of Machine Tool 4.0 63, p. 70.
- [2] Negi, P.K., Ram, M., Yadav, O.P., 2022. CNC Machine Programming Codes (G-Codes and M-codes), in *Basics of CNC Programming*, River Publishers, New York, p. 73.
- [3] Zhang, L., Liu, J., Zhuang, C., 2024. Digital Twin Modeling Enabled Machine Tool Intelligence: A Review 37.
- [4] Norda, M., Engel, C., Rennies, J., Appell, J.-E. et al., 2024. Evaluating the Efficiency of Voice Control as Human Machine Interface in Production 21, p. 4817.
- [5] Lotti, G., Villani, V., Battilani, N., Fantuzzi, C., 2019. New trends in the design of human-machine interaction for CNC machines 52, p. 31.
- [6] Börold, A., Schweers, D., Freitag, M., 2023. Towards Multimodal Information Systems for Assisting Humans in Production and Logistics Processes 120, p. 1089.
- [7] Kusuma, G., Sharmina, M., Gallego-Schmid, A., 2025. Decarbonising the electricity sector in Indonesia: Contradictions in national policies 135, p. 1113.
- [8] Abdelaal, M., Lokadjaja, S., Engert, G. GLLM: Self-Corrective G-Code Generation using Large Language Models with User Feedback.
- [9] Jayanthi, S., M, S., S, S.K., A, M.S. et al., 2025. Generative AI Solution for CNC Machines and Robotics Code Generation, in *2025 International Conference on Computational Innovations and Engineering Sustainability (ICCIES)*, IEEE, p. 1.
- [10] Hoang, D., Gorsich, D., Castanier, M.P., Imani, F., 2025. Knowledge Graph Fusion with Large Language Models for Accurate, Explainable Manufacturing Process Planning.
- [11] Ouerghemmi, C., Ertz, M., 2025. Integrating Large Language Models into Digital Manufacturing: A Systematic Review and Research Agenda 14, p. 318.
- [12] Striffler, N., Voigt, T., 2023. Concepts and trends of virtual commissioning – A comprehensive review 71, p. 664.
- [13] Sun, X., Zhang, F., Niu, X., Wang, J., 2024. A digital twin commissioning method for machine tools based on scenario simulation 77, p. 697.
- [14] Algarny, K., Abdelrahman, A.S., Youssef, M. A novel platform for power train model of electric cars with experimental validation using real-time hardware in-the-loop (HIL): A case study of GM Chevrolet Volt 2 nd generation.
- [15] Longo, F., Padovano, A., 2020. Voice-enabled Assistants of the Operator 4.0 in the Social Smart Factory: Prospective role and challenges for an advanced human-machine interaction 26, p. 12.
- [16] Ludwig, H., Schmidt, T., Kühn, M., 2023. Voice user interfaces in manufacturing logistics: a literature review 26, p. 627.
- [17] Wellsandt, S., Foosherian, M., Thoben, K.-D., 2020. Interacting with a Digital Twin using Amazon Alexa 52, p. 4.
- [18] Taulli, T., Deshmukh, G., 2025. Introduction to LangGraph, in *Building Generative AI Agents*, Apress, Berkeley, CA, p. 209.
- [19] Häfner, V., 2019. PolyVR - A Virtual Reality Authoring Framework for Engineering Applications. Karlsruhe.
- [20] Moshkovich, D., Zeltyn, S., 2025. Taming Uncertainty via Automation: Observing, Analyzing, and Optimizing Agentic AI Systems. arXiv.
- [21] Lambert, N., 2025. Reinforcement Learning from Human Feedback. arXiv.