
A Cognitive-Robotics Case Study in Autonomous Sanding

Moritz Schmidt 

MORITZ.SCHMIDT@THA.DE

Technical University of Applied Sciences Augsburg, Faculty of Electrical Engineering, Germany

Abstract

In our system, a Soar cognitive architecture instance acts as a cognitive control unit: it maintains an explicit task-decomposition model of a robotic sanding application and invokes neural models – currently a fine-tuned object-detection and pose-estimation pipeline – only as tools for the perception subtasks they perform best. The Soar-ROS 2 bridge, *soar_ros*, is open source and separately evaluated. The current work-in-progress version is a semi autonomous agent gated by human-in-the-loop validation steps before irreversible sanding actions. A planned lab study will add an explanation-chat interface to the same system and measure its effect on shop-floor workers’ attitude towards robots under controlled failure conditions. Post experiment interviews will explore the perceived quality of the explanations and use of the system.

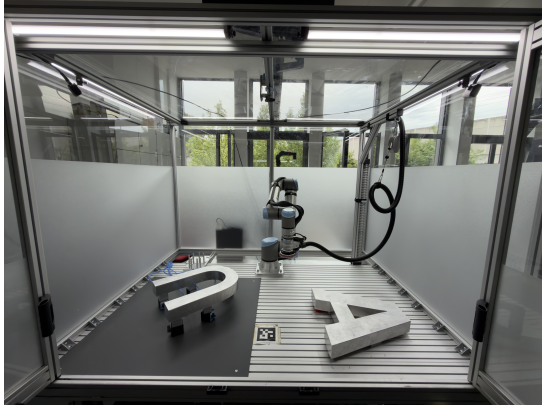
1. Implemented System: Soar as Cognitive Control Unit

Our system instantiates the OntoAgentic stance directly: a Soar (Laird, 2012) agent holds an explicit task-decomposition model of a robotic sanding application, use case details and previous work see (Schmidt & Meitinger, 2025) and (Schmidt et al., 2026) (Soar-ROS 2 bridge), orchestrating subtasks and invoking neural models only where they are the better tool. The perception is delegated to a fine-tuned object-detection tool (Straub et al., 2025). The current work-in-progress version is a semi autonomous agent gated by human-in-the-loop validation checkpoints before each irreversible sanding action (Figure 2).

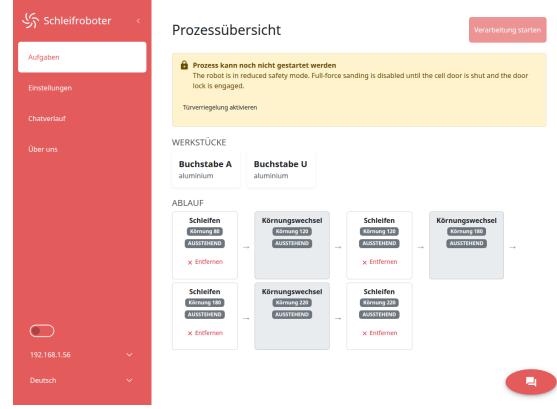
2. Upcoming Study: Explanations for the Autonomous Sanding Agent

Building on the autonomous, human-in-the-loop-gated system, we plan a qualitative, single-group lab study with shop-floor participants. A tablet-based chat interface (Figure 1) on the demonstrator lets operators query the running agent; we deliberately withhold information from the introduction material so that reproducible silent (e.g., an incomplete grit stock) and loud (e.g., a blocked precondition check) failures occur, letting us observe detection and recovery with and without explanations. Attitude toward robots is measured pre/post with ACIR-Q (Leichtmann et al., 2023), explanation quality with an in-interface satisfaction scale (Hoffman et al., 2023), alongside logs and a closing interview.

As shown in Figure 2, an MCP server exposes Soar’s working memory, see (Wray et al., 2026), and task decomposition over Soar’s SML interface as callable tools; the cloud LLM behind the tablet chat queries these on demand and composes its own natural-language answer, rather than following a fixed template. This differs from Kirk & Laird (2019) and Kirk (2026), who parse



(a) Sanding cell.



(b) Tablet UI.

Figure 1: The demonstrator used in the planned study.

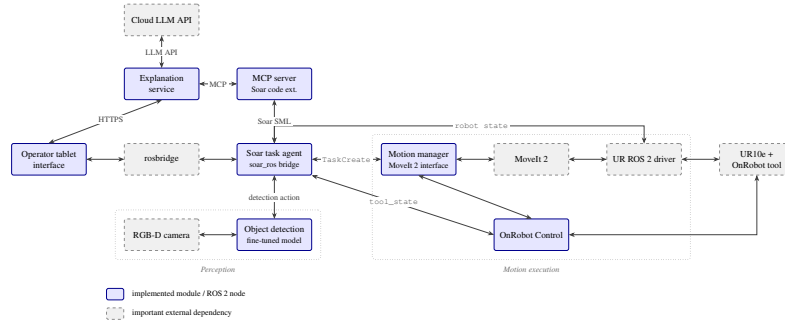


Figure 2: Planned system architecture (blue: implemented modules/ROS 2 nodes; grey: external dependencies). The explanation service reaches the Soar task agent out of band over SML via an MCP server and is the only component calling the cloud LLM.

a user question into a hierarchical logic graph (HLG) via a LLM, have Soar fill that HLG from memory, and pass the filled HLG to a second LLM call that composes the answer. Our approach skips the HLG creation: the LLM queries Soar’s working memory directly via MCP and composes its own answer without an intermediate structured representation – reconciling the two strategies is, we think, an open question for OntoAgentic explanation generation.

3. Challenging Problem

Soar represents working memory via a custom datamap format rather than an established ontology standard, limiting scalability and rule transfer across architectures. We pose grounding architectures’ datamaps in shared ontology formalisms as a concrete open question for the OntoAgentic AI community. At least Soar also faces old tooling, partially tried to resolve via (Wray et al., 2026; Schmidt et al., 2026).

Acknowledgements

Created as part of the AI Production Network Augsburg research project funded by StMWK (H.6-M2126.5.2.2.1/1/1/1) at THA.

References

- Hoffman, R. R., Mueller, S. T., Klein, G., & Litman, J. (2023). Measures for explainable AI: Explanation goodness, user satisfaction, mental models, curiosity, trust, and human-AI performance. *Frontiers in Computer Science*, 5, 1096257.
- Kirk, J. (2026). Hierarchical logic graphs. Presented at the 46th Soar Workshop.
- Kirk, J. R., & Laird, J. E. (2019). Learning hierarchical symbolic representations to support interactive task learning and knowledge transfer. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence* (pp. 6095–6102). Macao, China: International Joint Conferences on Artificial Intelligence Organization.
- Laird, J. (2012). *The Soar cognitive architecture*. MIT Press.
- Leichtmann, B., Hartung, J., Wilhelm, O., & Nitsch, V. (2023). New short scale to measure workers' attitudes toward the implementation of cooperative robots in industrial work settings: Instrument development and exploration of attitude structure. *International Journal of Social Robotics*, 15, 909–930.
- Schmidt, M., & Meitinger, C. (2025). Leveraging agent-based reasoning for natural task delegation on the shop floor. In *Studies in computational intelligence*, 41–54. Cham: Springer Nature Switzerland.
- Schmidt, M., Teynor, A., & Meitinger, C. (2026). soar_ros: A ros 2 integration of the soar cognitive architecture with performance evaluation and design guidelines. In *Service oriented, holonic and multi-agent manufacturing systems for industry of the future (sohoma)*. Springer.
- Straub, F., Schmidt, M., & Dietrich, S. (2025). Leveraging hybrid and synthetic data for object classification and detection: CNN vs. transformer architectures in autonomous robotic sanding. *2025 IEEE 7th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)* (pp. 1–8). Hamburg, Germany: IEEE.
- Wray, R. E., Schmidt, M., & Jones, S. J. (2026). Developer-on-the-loop: Engineering cognitive systems with coding agents. From <https://hal.science/hal-05691375>. Pre-print, ACS 2026.