Chair of Computing in Civil and Building Engineering
TUM School of Engineering and Design
Technical University of Munich

Master Thesis

LLM-driven Automated Compliance Checking by Advancing CodeAct Strategies

Description

Automated compliance checking (ACC) of building designs against relevant codes and standards has been an active research area for over 50 years but remains an open challenge. The complexity arises from multiple factors, including the intricacies of legal texts, the required domain knowledge, the subjective nature of certain regulations, variations in modeling practices, and the availability and completeness of building design data.

Following Eastman et al. (2009), the first step involves interpreting building regulations and converting them into a machine-readable format. Next, building design information must be extracted and enriched with additional data necessary for compliance verification. The reasoning process requires aligning regulatory terminology with design data and performing a range of compliance checks, from basic existence or quantitative tests to more advanced calculations, simulations, or qualitative assessments. Finally, the results must be documented alongside relevant building elements and their properties.
Recent advancements in Large Language Models (LLMs) offer new possibilities for compliance checking. Ying and Sacks (2024) demonstrated that LLMs can perform simple compliance checks by taking building regulations in natural language as input. The LLM invokes predefined functions to retrieve necessary design data and execute the required checks. While effective, this approach was limited by the predefined function set and was only tested on relatively simple regulations. Guo et al. (2025) and Iversen and Huang (2026) have demonstrated that LLMs are capable of aligning a large number of predefined functions with an input query (e.g., the regulatory clause) and can also generate implementations of those functions.

The integration of coding and reasoning capabilities in LLMs has been formalised in the CodeAct paradigm (Wang et al., 2024), which has proven effective for interacting with complex data sources and performing intricate analyses. We have successfully applied similar strategies to Automated Compliance Checking by generating Python code that interacts with IFC models via the IfcOpenShell library. This research will build on these efforts by investigating more advanced CodeAct strategies, including treating the Python environment as a stateful interactive environment, storing context in variables, and spawning new LLMs as subtasks (Zhang et al., 2025). Another strategy will be to investigate the impact of high-level LLM tools acting on different data formats and modalities (e.g., graphs, IfcJson, floorplans) and the use of specialized libraries (e.g., Trimesh, PythonOCC).

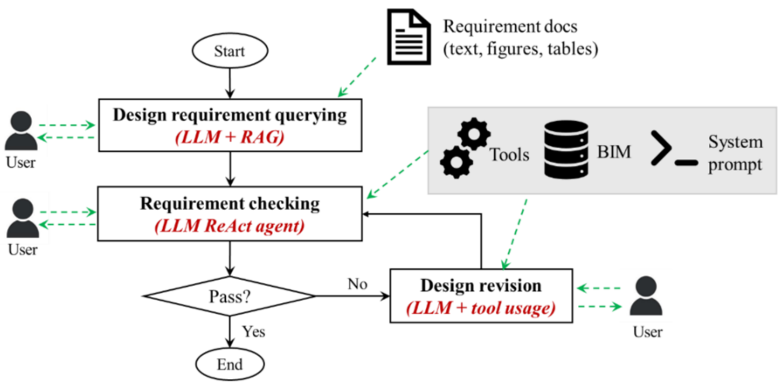

Figure 1. Agent-based ACC system by Ying and Sacks (2024)

Task

Investigate and compare various tools and approaches to LLM-driven ACC using the CodeAct paradigm.

- Conduct a literature review to identify existing challenges and solutions for LLM-driven ACC, as well as how advanced CodeAct strategies can bridge current research gaps.
- Develop an ACC dataset, potentially utilizing existing tools, such as Solibri Model Checker
- Develop a CodeAct-based coding agent to establish a baseline
- Investigate the impact of various enhancements, including but not limited to Graph-RAG, IFC file search, Read–eval–print loop (REPL), and Geometry libraries

Prerequisites

Necessary: Programming skills (Python)
Good to have: Experience with LLMs and developing LLM agents, Machine learning frameworks, BIM, IFC, IfcOpenShell, Computational geometry

Supervisors

Stefan Fuchs, Chair of Computing in Civil and Building Engineering, stefan.m.fuchs@tum.de
Sylvain Hellin, Chair of Computing in Civil and Building Engineering, sylvain.hellin@tum.de

References

Eastman, C., Lee, J.-M., Jeong, Y.-S., & Lee, J.-K. (2009). Automatic rule-based checking of building designs. *Autom. Constr.*, *18*(8), 1011–1033. https://doi.org/10.1016/j.autcon.2009.07.002

Guo, P., Xue, H., Ma, J., & Cheng, J. C. P. (2025). Advancing BIM information retrieval with an LLM-based query-domain-specific language and library code function alignment system. *Autom. Constr.*, *178*(106374), 106374. https://doi.org/10.1016/j.autcon.2025.106374

Iversen, O., & Huang, L. (2026). Leveraging large language models for BIM-based automated compliance checking. *Automation in Construction*, *182*, 106707. https://doi.org/10.1016/j.autcon.2025.106707

Wang, X., Chen, Y., Yuan, L., Zhang, Y., Li, Y., Peng, H., & Ji, H. (2024). Executable code actions elicit better LLM agents. In *arXiv [cs.CL]*. http://arxiv.org/abs/2402.01030

Ying, H., & Sacks, R. (2024, November 5). *From Automatic to Autonomous: A Large Language Model-driven Approach for Generic Building Compliance Checking*. Proceedings of the 41st International Conference of CIB W78, Marrakech, Morocco, 2-3 October, ISSN: 2706-6568. http://itc.scix.net/paper/w78-2024-59

Zhang, A. L., Kraska, T., & Khattab, O. (2025). *Recursive Language Models* (No. arXiv:2512.24601; Version 1). arXiv. https://doi.org/10.48550/arXiv.2512.24601