Static analysis of concurrency - algorithms vs AI

Short Description

LLMs can offer assistance in different areas of life. In engineering, LLMs are still observed with skepticism considering their issues with the reproducibility of the results and general imprecision. However, in software engineering, certain problems are undecidable. Static analysis in certain programming languages cannot statically evaluate program state and is hence forced to make over proximations. This results in loss of precision of such analysis.
In concurrent software systems with multiple threads, it is extremely hard to statically reconstruct shared states and synchronization mechanisms (especially in languages such as C and C++). This project aims to employ several LLM tools and, through a case study, conclude what is the potential of these tools to complement static analysis. As part of the project, the candidate will also develop algorithms for static analysis (e.g., Eraser Lockset [1]) and use existing tools (e.g., ThreadSanitizer [2]) to compare their results with the LLM tools.

[1] Eraser: A dynamic data race detector for multi-threaded programs – Savage et al. 1997
[2] https://github.com/google/sanitizers