

# A Visualization Interface for GenMC

## Thesis Description

Student: Valon Lusic, 22-830-920

Advisors: Prof. Dr. M. Kokologiannakis, M. Roshardt

## Summary

GenMC is a stateless model checker for C/C++ programs written under a variety of weak memory models. In such models, the executions of concurrent programs are represented as execution graphs, which are comprised of a set of events (graph nodes) that stand for individual memory accesses performed in the program, and relations on the events (graph edges). GenMC's verification process involves generating execution graphs of a program, by adding events incrementally and checking for safety violations at every step. Whenever a safety violation occurs, a text-based error report is created. This report prints an offending execution graph and highlights event(s) that caused the violation.

GenMC's text-based output for a given two-dimensional graphical structure is counter-intuitive. A visualization interface that represents the reported graph in its natural 2-D form would benefit GenMC's debugging and error reporting a lot. The creation of such a visualization interface is the main objective of this project.

The visualizations will be built using text output of GenMC as input. The resulting static and potentially interactive interface should present the exploration process in a clear, user-friendly way, offering an overview via expandable or collapsible tree structures and filtering mechanisms that highlight relevant information. This kind of tool will help developers and researchers the most, as it will visualize the corresponding execution graph whenever something unexpected (e.g., a bug) occurs.

Additionally, the tool should visualize GenMC's exploration algorithm, to help researchers who are learning about GenMC better understand how the verification process works under the hood.

## Project Goals

- **Core goals**
  - Gain familiarity of the algorithm (TruST) at the core of GenMC.
  - Survey and choose appropriate technologies for the project.
  - Provide a visualization of GenMC's exploration process, based on GenMC's current text output printing structure, additional assumptions, and API constraints.
  - Implement a static visualization interface.
- **Extension goals**

- Add interactivity to the tool, allowing interactive exploration of GenMC's output, such as step-by-step inspection of exploration paths or exports of exploration paths as vector graphics (e.g., SVG or PDF) or as a bitmap (e.g., PNG).
- Add a backend to export exploration graphs into .dot or LaTeX format, allowing users to produce printable representations of exploration paths.
- Investigate and document potential extensions of GenMC to further better the visualization interface.

## Timeline

- **Initial phase (1–2 weeks)**
  - Thoroughly read the paper "Truly Stateless, Optimal Dynamic Partial Order Reduction" (2022).
- **Requirement analysis (1–2 weeks)**
  - Understand what GenMC currently prints out, existing export functionalities, and which outputs/logs are reliable for visualization.
- **Prototype implementation (3–4 weeks)**
  - Parse GenMC's output/logs.
  - Build a static version of the visualization interface using a language with existing visualization libraries.
- **Extension development (1–2 weeks)**
  - Add optional interactive features to the interface.
  - Implement graph export functionality (e.g., .dot, LaTeX).
- **Final phase (4 weeks)**
  - Write the thesis.
  - Final refinements of the interface.