

## Bachelor/Master Thesis

### LLVMTA: Utilising linked binaries for analysed address space

Nils Hölscher  
 Prof. Dr. Jian-Jia Chen  
 Otto-Hahn Str. 16  
 Technische Universität Dortmund  
 Email: [nils.hoelscher@tu-dortmund.de](mailto:nils.hoelscher@tu-dortmund.de)  
 10.03.2022

LLVMTA is a static Worst Case Execution Time (WCET) analyser published recently. Such analysers determine an upper bound of the program's execution time (so-called WCET), which is very useful for real-time systems. To estimate the WCET of a program, its Control Flow Graph (CFG) has to be reconstructed from the binary, by which all necessary analyses approaches can be conducted, like value analysis, path-analysis and cache analysis, as shown in Figure 1.

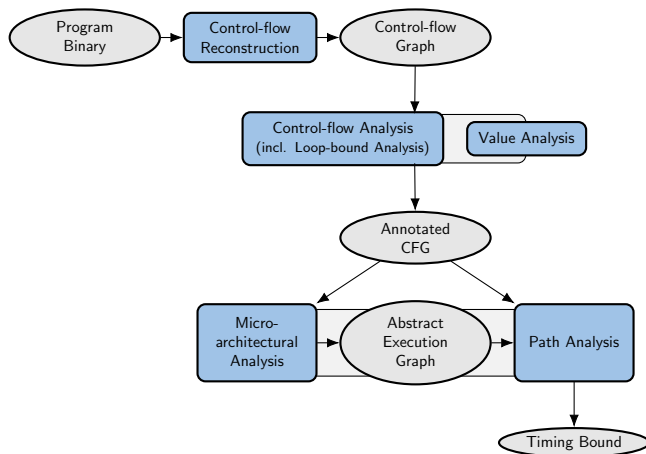


Figure 1: Overview of the general steps in WCET analysis.

Conventional tools like OTAWA or aiT take the generated binary code to reconstruct the CFG. However, it is already generated during a programs compilation, and the reconstructed one might lose important information for analysis.

LLVMTA takes a different approach: As it is integrated into LLVM, the program can be analysed from its C code, instead of the binary representation. The whole compilation flow of LLVM and LLVMTA is shown in Figure 2. However, address mapping, indicated by the dotted line in Figure 2, is not implemented yet. This means LLVMTA does not use the actual address mapping of the compiled binary but a rough estimate of it. Making the whole result unsafe.

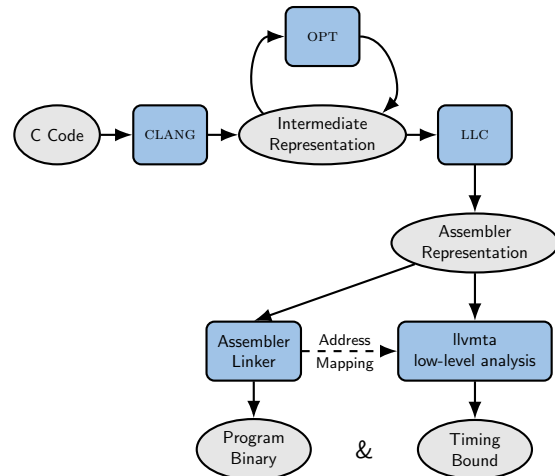


Figure 2: Overview of the common LLVM compilation flow (CLANG, OPT, LLVM) including the integration of our low-level analysis tool LLVMTA.

In this thesis, students should make themselves comfortable with LLVMTA and LLVM. While doing so, the two methods of linking a binary and using its address mapping, should be compared.

1. Linker like LLD + object dump.
2. LLVMs JIT (Just In Time) framework.

One of these methods should be implemented into LLVMTA for at least RISC-V or ARM32.

Other suggestions and related topics are also welcome. Please do not hesitate to make an appointment.

#### Required Skills:

- Knowledge of computer architecture
- Basic/Advanced knowledge of C++

#### Acquired Skills after the thesis:

- Insights into the LLVM compiler ecosystem
- Knowledge about static WCET analysis.

#### Literature:

- [LLVMTA: An LLVM-Based WCET Analysis Tool](#)