Exact Loop Bound Synthesis

Keywords: Binary analysis, artificial intelligence, program synthesis

Main Supervisor
Grégoire MENGUY
CEA LIST
gregoire.menguy@cea.fr

Senior Supervisor
Sébastien BARDIN
CEA LIST
sebastien.bardin@cea.fr

Many program analyses try to recover the number of loop iterations. It helps code verification, enables comparing the execution cost of software, or even enables assessing the conditional equivalence of execution costs [7]. Unfortunately, most analyses do not compute the exact number of loop iterations but an upper bound, which is not precise enough in many contexts. Recently, new approaches have been proposed to compute the exact number of loop iterations [7]. These rely on program synthesis, i.e., algorithms that generate a code snippet from a user-given specification.

On the other hand, a wide variety of program synthesizers have been proposed [2, 1, 5, 4]. Most target a specific use case, from Excel MACRO synthesis [4] to binary code deobfuscation [6, 3]. Especially, the open source Xyntia synthesizer from the CEA (`https://github.com/binsec/xyntia`) was shown to be very efficient on deobfuscation [6].

In this project, we aim to extend the Xyntia synthesizer to tackle the *exact loop bound synthesis problem*. It will proceed as follows:

- Get familiar with the state-of-the-art of program synthesis and exact loop bound analysis;
- Extend the Xyntia synthesizer to tackle the exact loop bound problem;
- Evaluate Xyntia over exact loop bound analysis tasks found in the literature;
- (Optional) Propose an extension in Xyntia to improve its performance.

1 Expected deliverable

- A summary of the bibliography made by the student;
- A documented implementation of the proposed extensions;
- A final report (slides) with the first two deliverables and a summary of the research and results.

2 Organization

Regular meetings will be organized with the supervisor (online or in person at CEA Nano-Innov, Saclay).

3 How to apply

To apply, students must send an email to the main supervisor (Grégoire Menguy) with the senior supervisor (Sébastien Bardin) in copy.

References

[1] Haniel Barbosa, Clark Barrett, and Martin Brain et al. "cvc5: A Versatile and Industrial-Strength SMT Solver". In: *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*. Springer, 2022.

[2] Clark Barrett, Christopher L. Conway, and Morgan Deters et al. "CVC4". In: *Conference on Computer Aided Verification (CAV '11)*. Springer, 2011.

[3] Tim Blazytko et al. "Syntia: Synthesizing the Semantics of Obfuscated Code". In: USENIX *Security*. 2017.

[4] José Cambronero, Sumit Gulwani, and Vu Le et al. "FlashFill++: Scaling Programming by Example by Cutting to the Chase". In: *Proc. ACM Program. Lang.* POPL (2023).

[5] Yuantian Ding and Xiaokang Qiu. "Enhanced Enumeration Techniques for Syntax-Guided Synthesis of Bit-Vector Manipulations". In: *POPL'24* ().

[6] Grégoire Menguy, Sébastien Bardin, and Bonichon et al. "Search-Based Local Black-Box Deobfuscation: Understand, Improve and Mitigate". In: *Conference on Computer and Communications Security*. 2021.

[7] Daniel Riley and Grigory Fedyukovich. "Exact Loop Bound Analysis". In: *Proceedings of the ACM on Programming Languages* 9.PLDI (2025), pp. 1814–1837.