

The research group
Software Languages Lab

has the honor to invite you to the public defence of the PhD thesis of

Noah Van Es

to obtain the degree of Doctor of Sciences

Title of the PhD thesis:

**Abstract Garbage Collection
for Efficient and Precise Abstract Interpretation**

Curriculum vitae

Supervisor:
Prof. dr. Coen De Roover (VUB)

The defence will take place on
Friday, October 23, 2026 at 4 p.m.

VUB Etterbeek campus, Pleinlaan 2, Elsene,
Promotiezaal D.2.01

The defence can be followed through a live
stream. A link can be requested upfront via
noah.van.es@vub.be.

Members of the jury

Prof. dr. Wolfgang De Meuter (VUB, chair)
Prof. dr. Jens Nicolay (VUB)
Prof. dr. Joost Vennekens (VUB)
Dr. Caterina Urban (INRIA Paris, FR)
Dr. Raphaël Monat (INRIA Lille, FR)

Noah Van Es obtained his MSc in Computer Science at the Vrije Universiteit Brussel, summa cum laude. He then started his PhD at the Software Languages Lab under the supervision of Prof. dr. Coen De Roover. His research interests mainly lie in static program analysis through abstract interpretation.

His work resulted in multiple peer-reviewed publications, including 7 as a first author, which he presented at several international conferences. In addition, he collaborated as a co-author on 4 other peer-reviewed publications. During his PhD, he also worked on the ECOPIPE and BUGATTI research projects, and guided 7 bachelor theses and 5 master theses.

Abstract of the PhD research

Abstract garbage collection (GC) is the application of garbage collection to an abstract interpreter. It is known that abstract GC significantly improves both the interpreter's precision and performance. In this work, we focus on the efficient integration of abstract GC.

In order to maximise its precision and performance benefits, abstract GC needs to be applied frequently; at the same time, frequent applications bring about their own performance overhead.

We propose a novel approach for abstract GC based on reference counting (abstract reference counting). The approach is able to immediately eliminate all garbage, maximising the precision and performance benefits of abstract GC. At the same time, it does not require frequent heap traversals, therefore adding no significant overhead. In order to ensure all garbage is eliminated, we augment abstract reference counting with cycle detection by counting references at the level of strongly connected components. The empirical evaluation shows that the resulting approach achieves both the optimal precision as well as significantly improved performance compared to existing approaches to abstract GC.

Second, we consider the integration of abstract GC for abstract interpreters that do not explicitly model the stack (such as abstract definitional interpreters). Existing techniques either fail to exploit the full benefit of both abstract GC and pushdown control flow, and/or require complicated modifications to the abstract interpreter. In addition, they all lack scalability, as abstract GC is not compatible with global store widening. We present delta store semantics (DSS), a novel yet simple approach to integrating abstract GC into abstract definitional interpreters. DSS returns a delta store (representing changes to the original store) instead of the updated store as part of the evaluation result. DSS not only preserves the advantages of pushdown control flow and abstract GC, but is able to exploit even greater precision benefits (due to more aggressive GC). We confirm this both formally as well as empirically. To prevent scalability issues, we propose a new form of store widening for DSS, which allows for both flow sensitivity as well as a (limited) form of abstract GC.

Finally, we show how an existing effect-driven worklist algorithm can be generalised to efficiently compute the fixpoint for DSS-based analyses (both with and without abstract GC). In order to further improve performance, we also design and implement a parallel version of this algorithm. Our empirical evaluation shows that this algorithm scales well on multi-core architectures for DSS-based analyses.