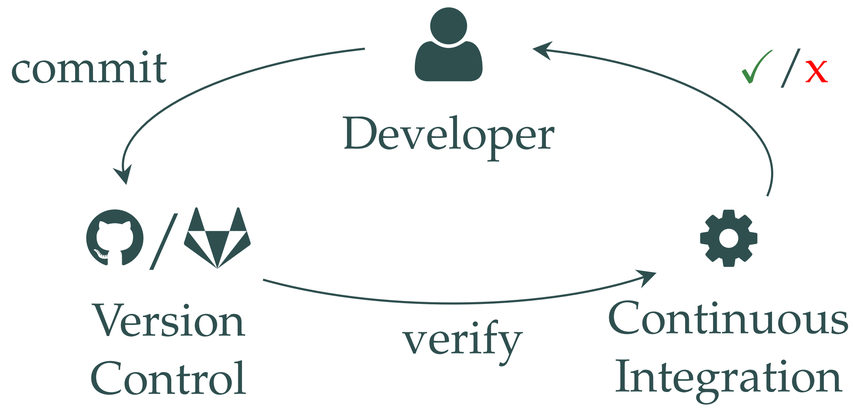
Regression Verification with Precision Reuse

Márk Somorjai
mark.somorjai@lmu.de
Supervisors: Marie-Christine Jakobs, Dirk Beyer
Collaborators: Marian Lingsch-Rosenfeld

Overview

Modern software development processes pose challenges for verification [3].



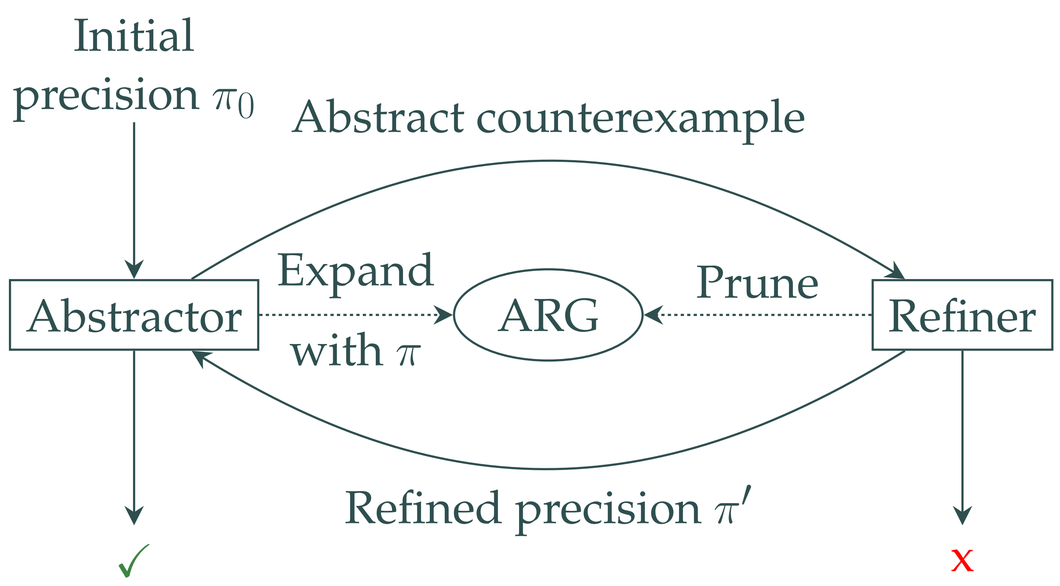
Verifying the whole program on each commit:

- wastes resources on unchanged parts
- does not adhere to fast response times

Idea: make verification incremental as well.

Abstraction

Counterexample-Guided Abstraction Refinement (CEGAR) [2] iteratively calculates the **precision**, i.e., the level of abstraction necessary for verification.

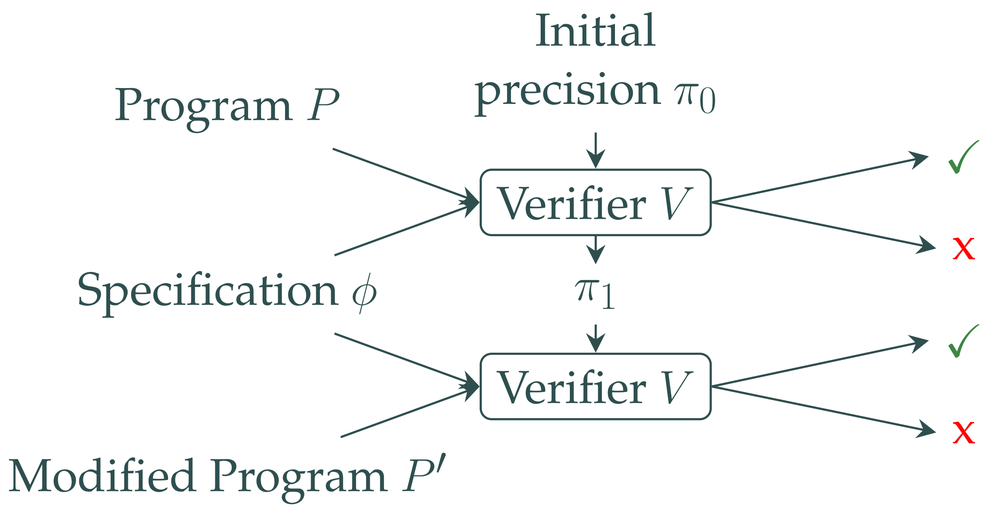


Precision depends on the abstract domain, e.g.:

- predicate abstraction: $\{x > 0, a + b = 2\}$
- value analysis: $\{x, a, b\}$

Precision Reuse

Proposed in 2013 [1] to reuse the calculated precisions between verification runs.



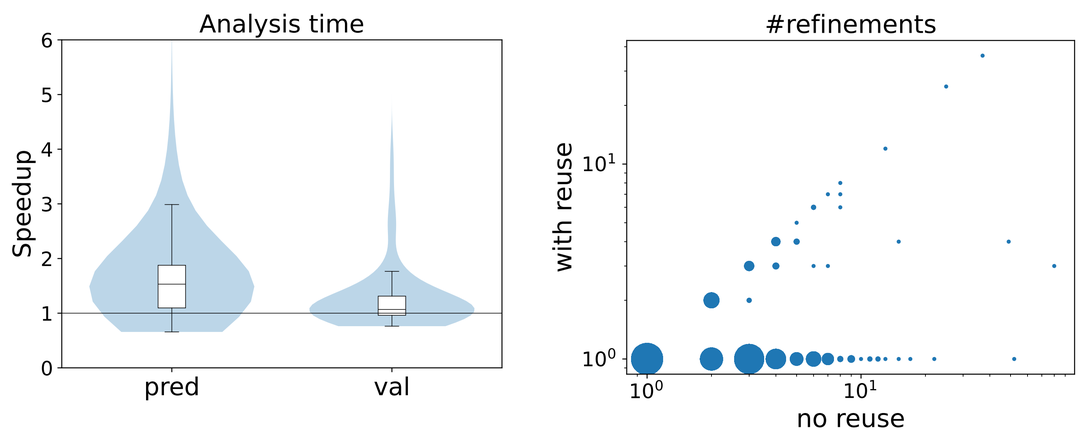
Advantages:

- only effects abstraction → maintains **soundness**
- fewer CEGAR iterations → **speedup**

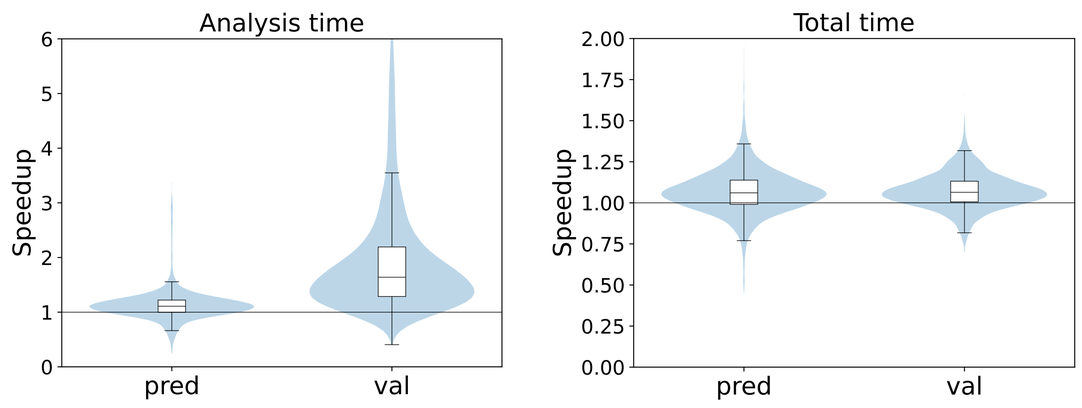
Evaluation

Reproduction & replication study on the benefits of precision reuse in today's verification landscape.

Replication: implementation in Otheta



Reproduction: new CPA✓ version



Systematic assessment of the original claims [1]:

Performance	Realizability
✓ improved effectiveness	✓ easy to implement
? less #refinements	✓ easy to serialize
? improved efficiency	? tool-independent
✓ low overhead	
? small precision files	

Key findings:

- **positive effects** are still **mostly observed**, but not in all cases anymore
- refinement time has been reduced proportionally → **smaller benefits**

Future work: exchange precisions with witnesses.

References

[1] D. Beyer et al. "Precision reuse for efficient regression verification". In: *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*. ESEC/FSE 2013. ACM, 2013, 389–399.

[2] E. Clarke et al. "Counterexample-Guided Abstraction Refinement for Symbolic Model Checking". In: *J. ACM* 50.5 (2003), 752–794.

[3] P. W. O'Hearn. "Continuous Reasoning: Scaling the impact of formal methods". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS '18. ACM, 2018, 13–25.