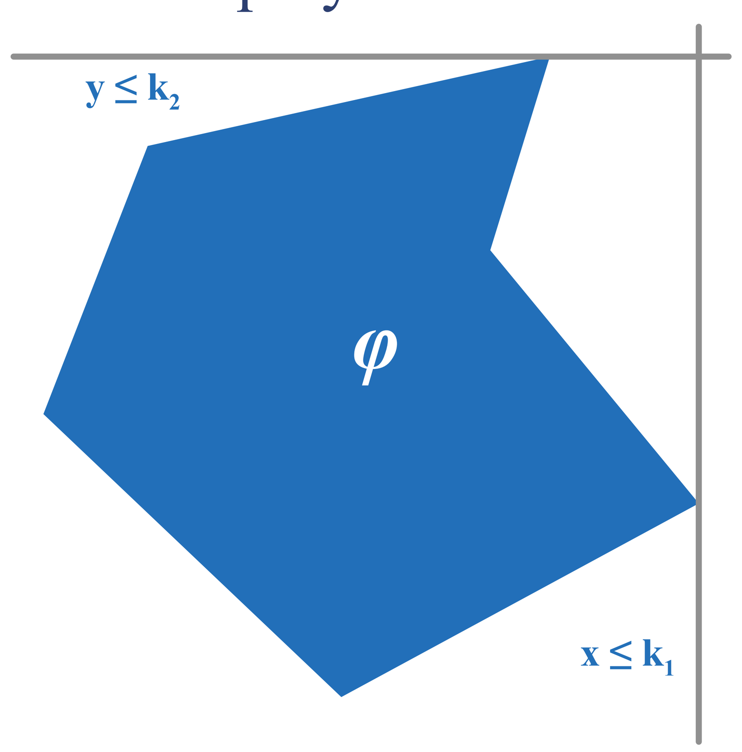
Symbolic Optimization with SMT Solvers

Yi Li[1], Aws Albarghouthi[1], Zachary Kincaid[1], Arie Gurfinkel[2], Marsha Chechik[1]

1. University of Toronto 2. Software Engineering Institute, CMU

Problem Statement

- ❑ Optimize linear objective functions ***T={t_1, …, t_n}*** subject to quantifier-free linear real arithmetic (QF_LRA) constraints ***φ***.
- ❑ Geometrically, find tightest bounds for non-convex polyhedron.



The problem is known as Symbolic Optimization.

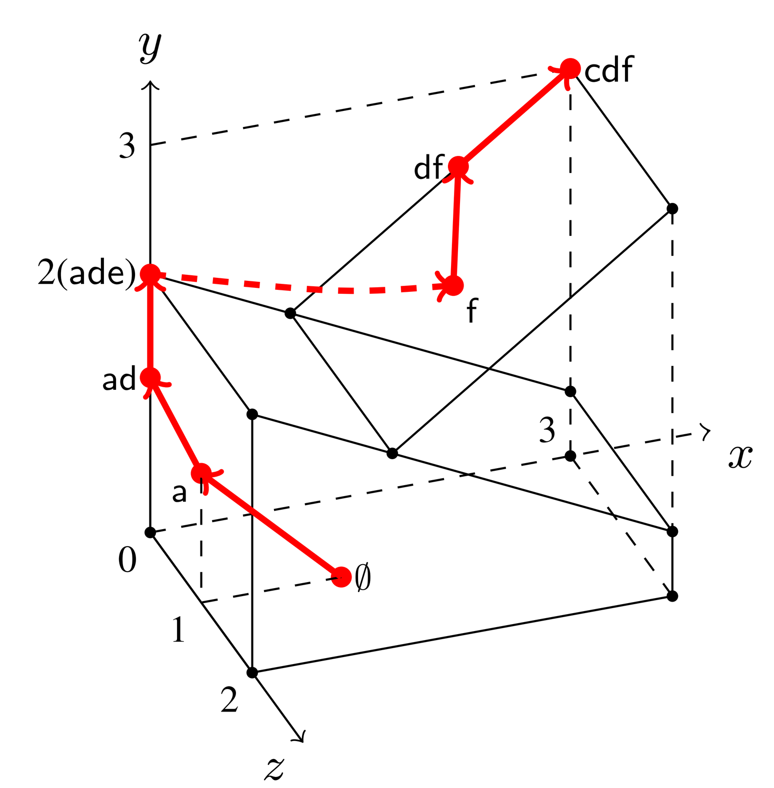
Applications in PL

- ❑ Numerical invariant generation: implementing the most precise abstract transformers [1] for various numerical abstract domains.
- ❑ Counterexample generation: finding optimal counterexamples that maximize/minimize certain criteria.
- ❑ Program synthesis: synthesizing programs with lowest costs in performance critical contexts.
- ❑ Constraint programming: extending constraint solvers [2] with the ability of returning optimal solutions.
- ❑ Interpolation generation: simplifying unsatisfiability proofs which can be used to generate simpler interpolants [3].

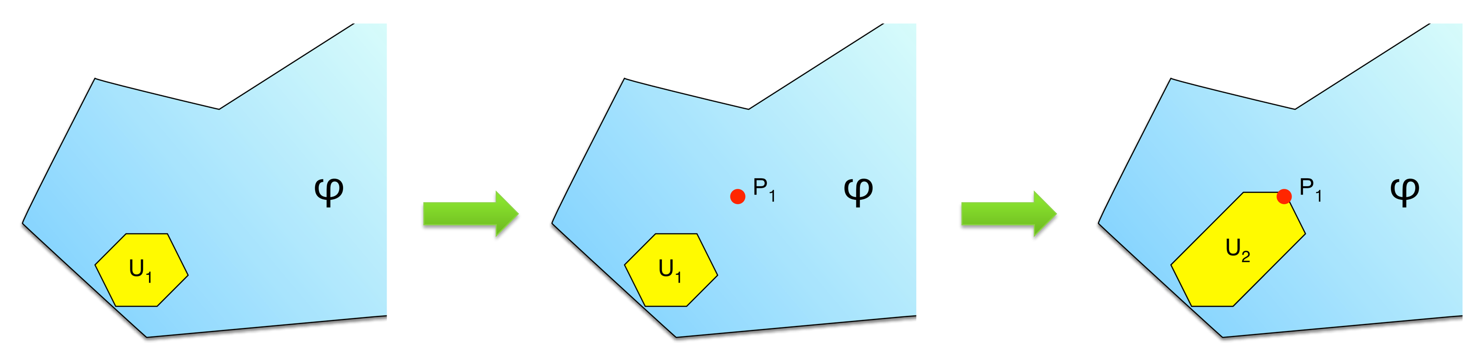
The SYMBA Approach

SYMBA maintains an under-approximation (***U***) of the optimal solution and grows ***U*** as a series of SMT-based sampling rule applications:

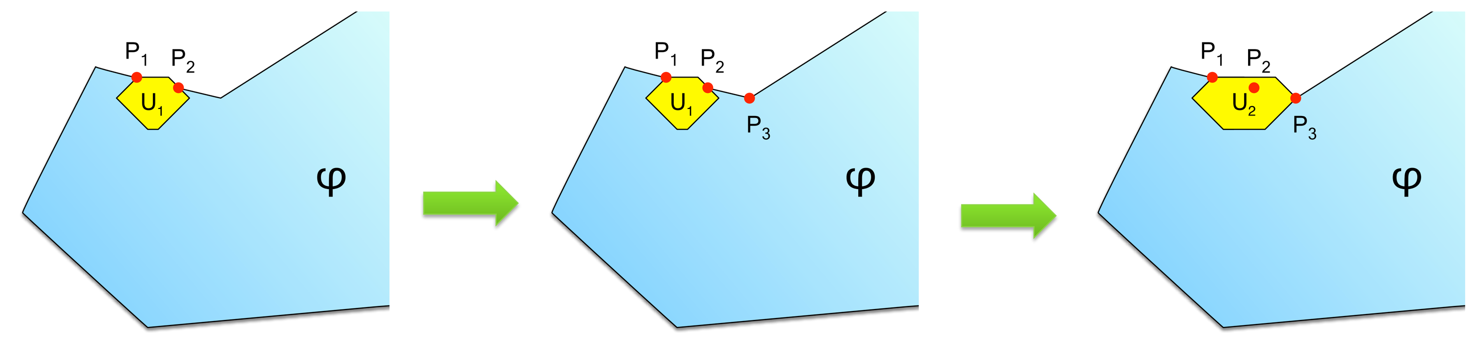
$\varphi: 0 \le x \le 3 \wedge 0 \le z \le 2 \wedge (2y \le -x+4 \vee 4y = 3x+3)$



- ❑ **GlobalPush (GP):** sample a point outside of ***U***.

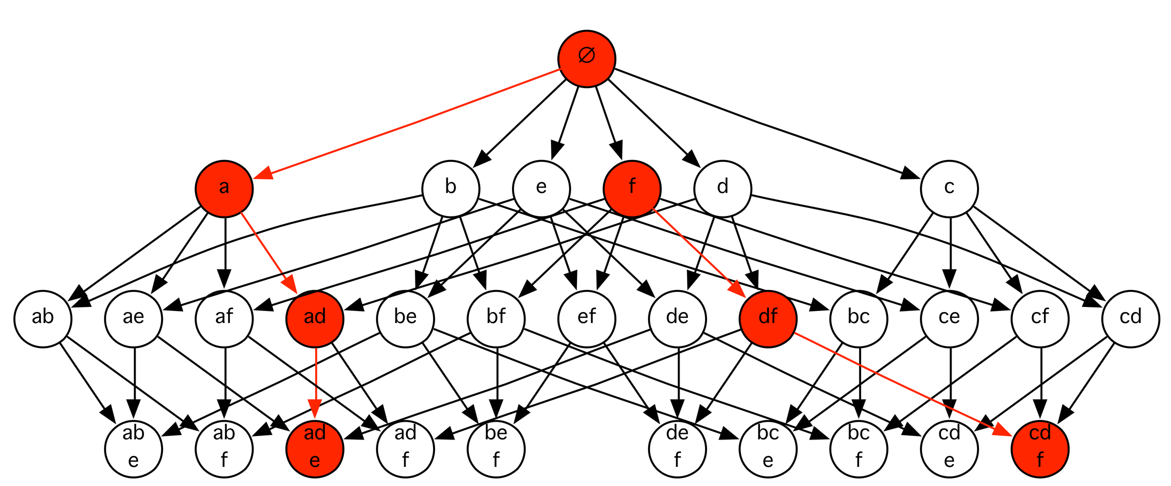


- ❑ **Unbounded (UB):** check unboundedness and sample a vertex as a side effect.



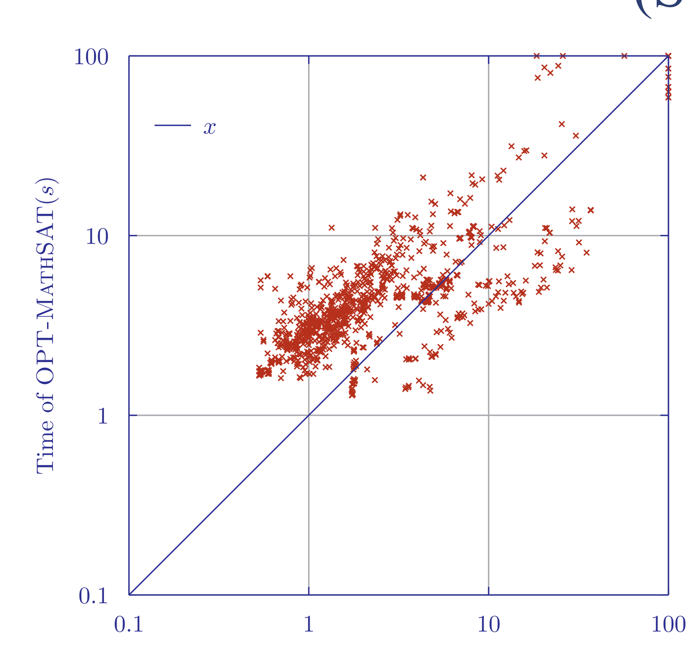
a: x=0
b: z=2
c: x=3
d: z=0
e: x+2y=4
f: -3x+4y=3

GP: ∅
UB: ∅ → a → ad → ade
GP: ade → f
UB: f → cf → cdf

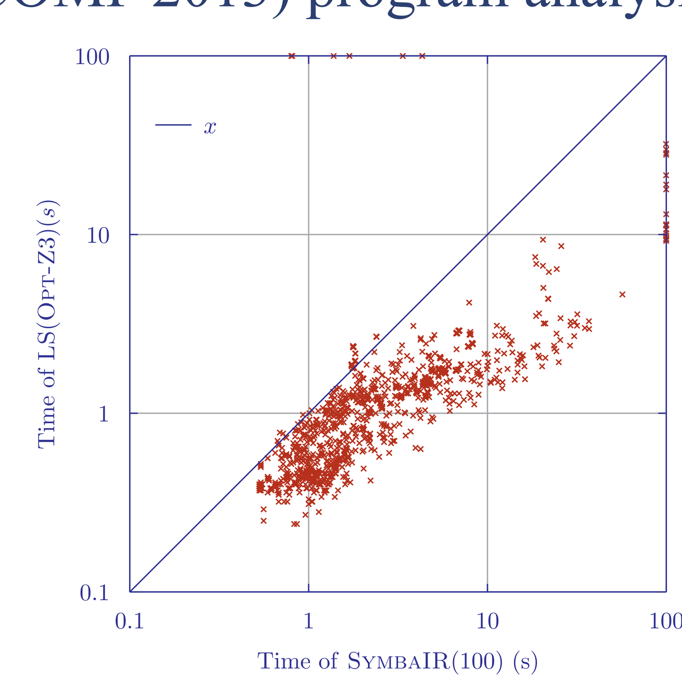


Experimental Evaluation

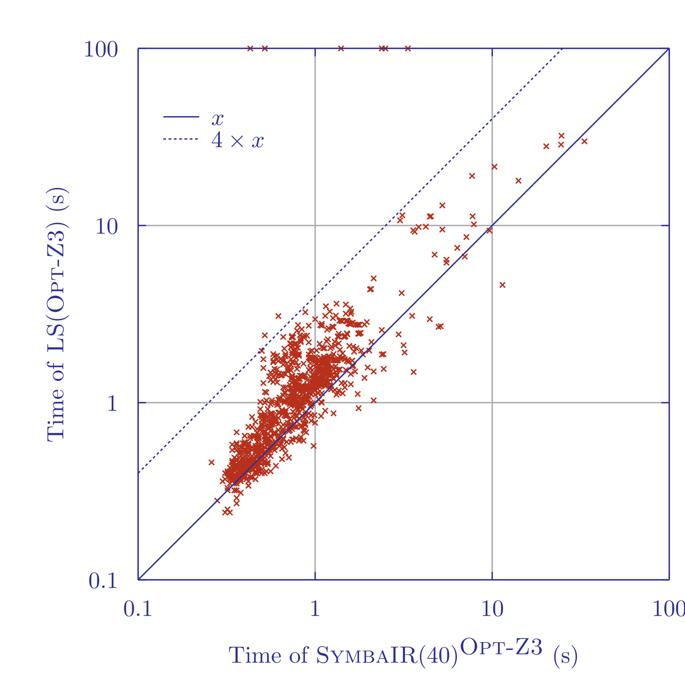
Performance comparisons on benchmark set obtained from Competition on Software Verification (SV-COMP 2013) program analysis tasks



SYMBA vs. OptMathSAT



SYMBA vs. LS(Opt-Z3)



SYMBA+LS(Opt-Z3) vs. LS(Opt-Z3)

Our Solution: SYMBA

SYMBA is a novel SMT-based optimization algorithm for objective functions in linear real arithmetic:

- ❑ Utilizes efficient SMT solvers as black boxes
- ❑ Handles a mix of different theories, e.g., array, Boolean, LRA
- ❑ Flexible and configurable algorithm that is easy to optimize
- ❑ Optimizes a set of objective functions, reusing information among them to speed up the optimization task
- ❑ Extensive evaluation against other proposed techniques on program analysis benchmarks
- ❑ Implementation and benchmarks are available at: http://bitbucket.org/arieg/ufo

Conclusion and Next Step

SYMBA solves the symbolic optimization problem by systematic and efficient point sampling via SMT queries. Experimental evaluation indicates advantages over other techniques. Future work:

- ❑ Extend to integer arithmetic
- ❑ Handle non-linear objective functions
- ❑ Exploit parallelism in implementation

References

[1] T. Reps, M. Sagiv, and G. Yorsh. Symbolic Implementation of the Best Transformer. In *Proc. of VMCAI'04*, volume 2937 of *LNCS*, 2004.

[2] A. S. Köksal, V. Kuncak, and P. Suter. Constraints as Control. In *Proc. of POPL'12*, pages 151–164, 2012.

[3] A. Albarghouthi and K. L. McMillan. Beautiful Interpolants. In *Proc. of CAV'13*, pages 313–329, 2013.