UFO: Verification with Interpolants and Abstract Interpretation

Arie Gurfinkel and Sagar Chaki
Software Engineering Institute
Carnegie Mellon University

Aws Albarghouthi, Yi Li and Marsha Chechik
University of Toronto

UFO

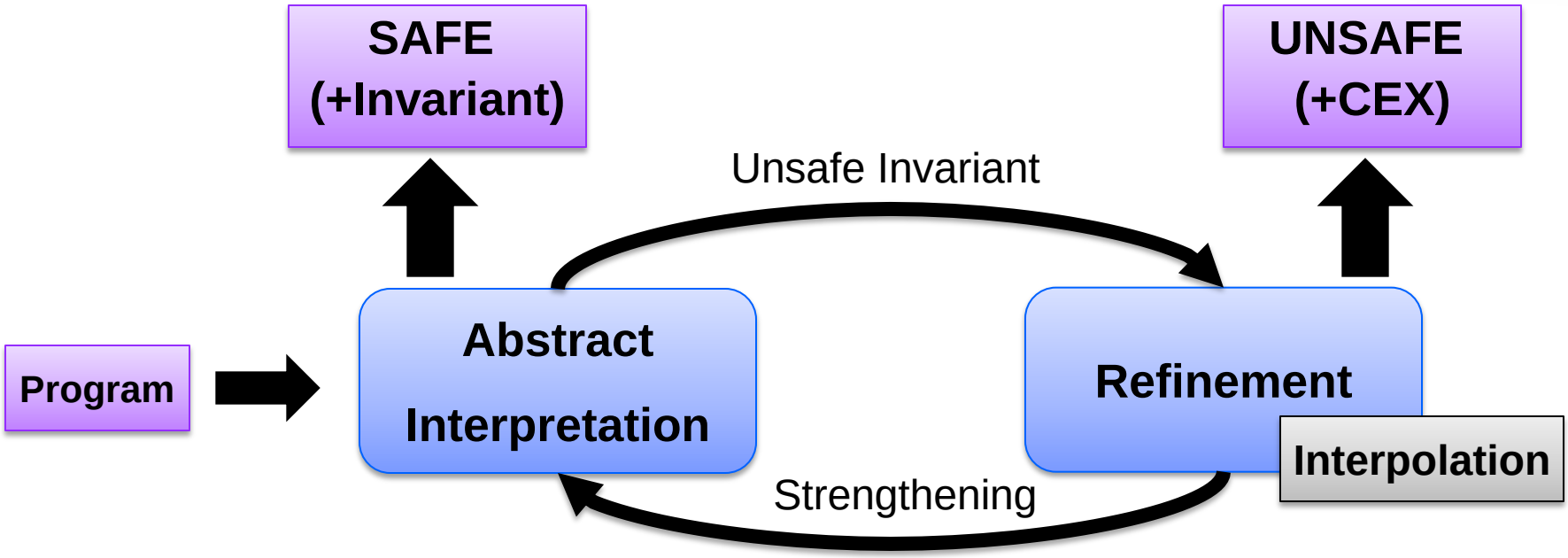
- A *framework* and a *tool* for software verification
- Tightly integrates *interpolation-* and *abstraction*-based techniques

Check it out at:
http://bitbucket.org/arieg/ufo

References:

[SAS12] Craig Interpretation
[CAV12] UFO: A Framework for Abstraction- and Interpolation-based Software Verification
[TACAS12] From Under-approximations to Over-approximations and Back
[VMCAI12] Whale: An Interpolation-based Algorithm for Interprocedural Verification

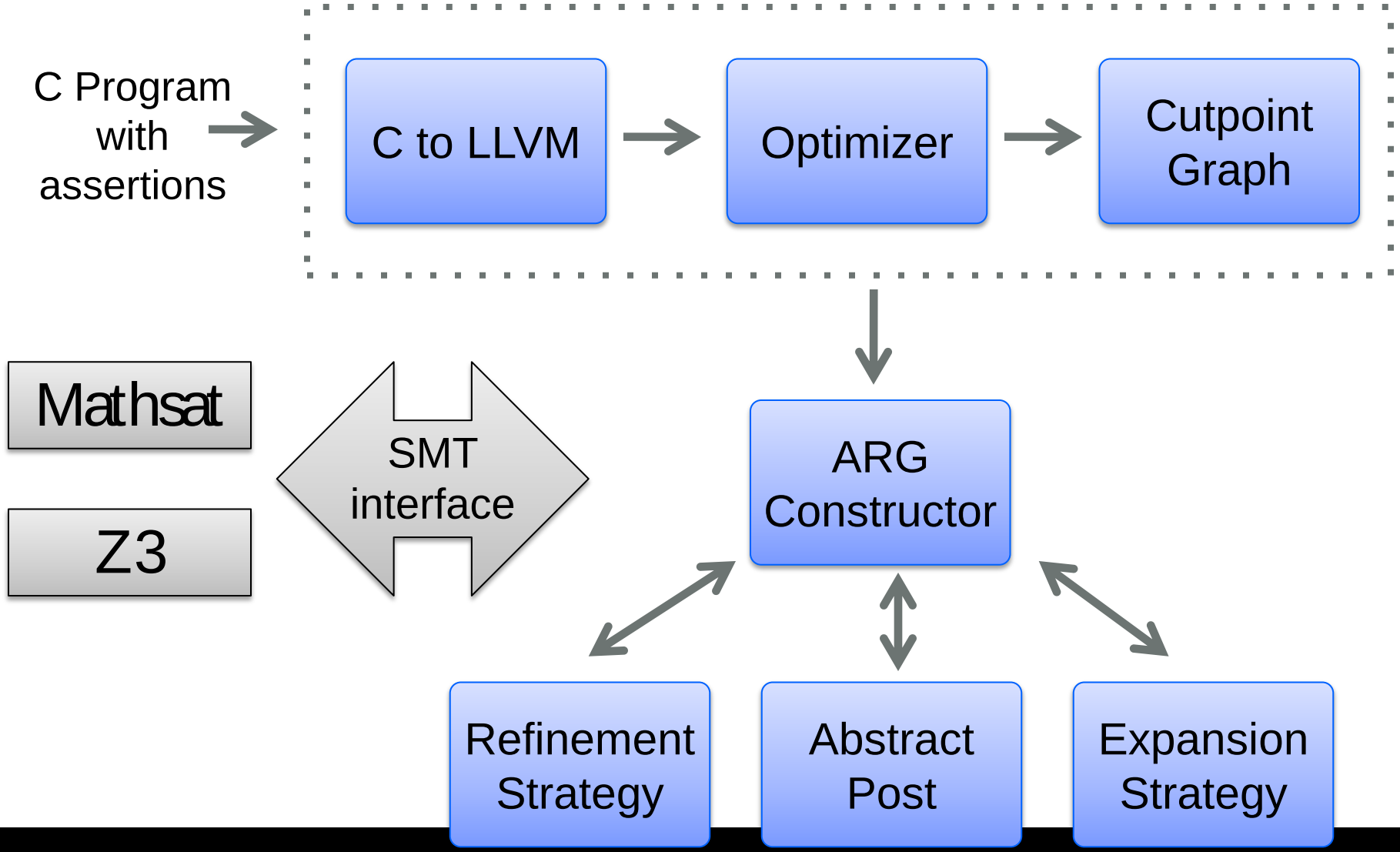
Verification with INTERP and AI



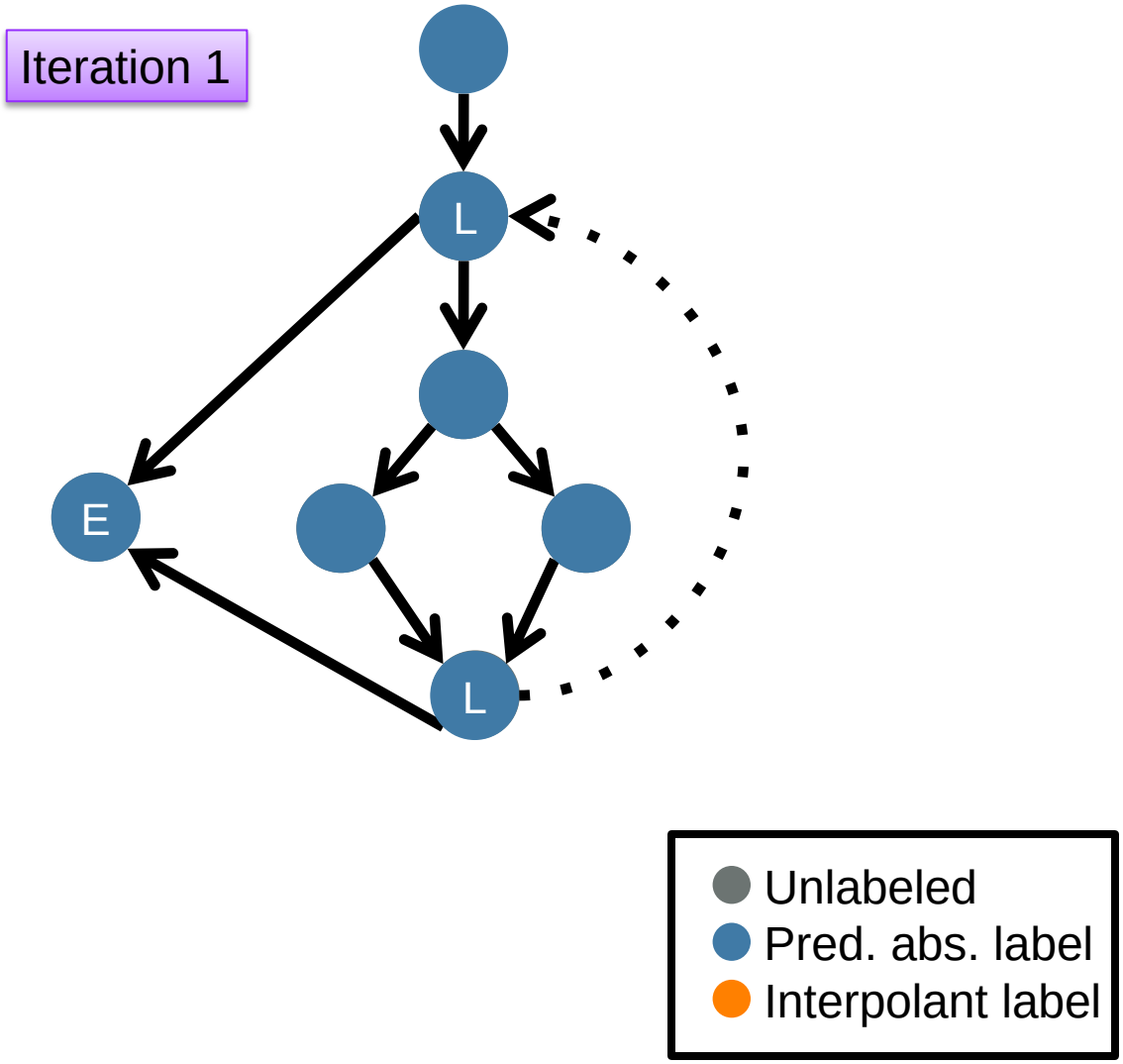
- uses Cutpoint Graph (CPG)
- maintains an unrolling of CPG
- computes disjunctive invariants
- uses novel powerset widening

- uses SMT to check for CEX
- DAG Interpolation for Refinement
- Guided by AI-computed Invs
- Fills in “gaps” in AI

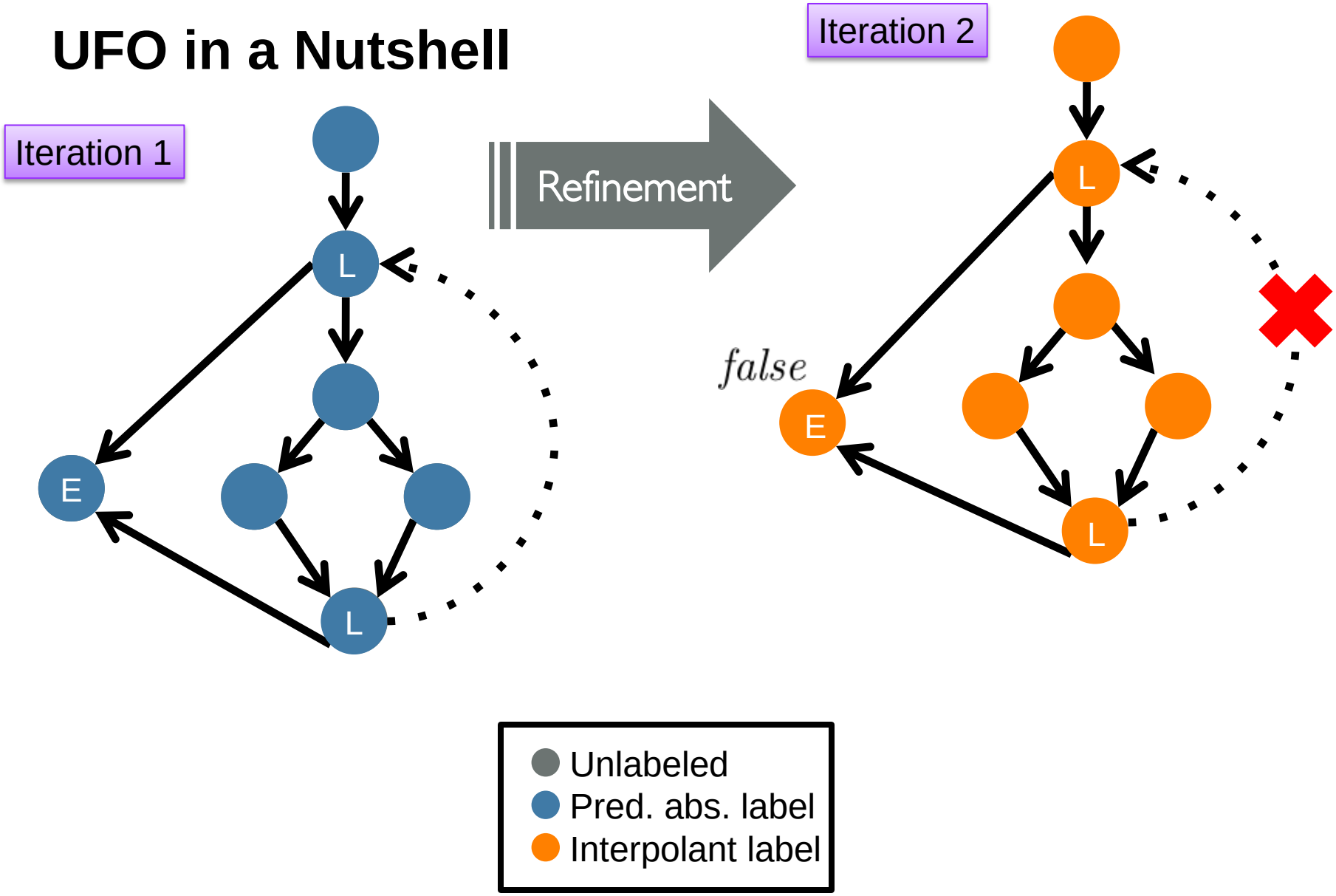
Implementation in UFO Framework



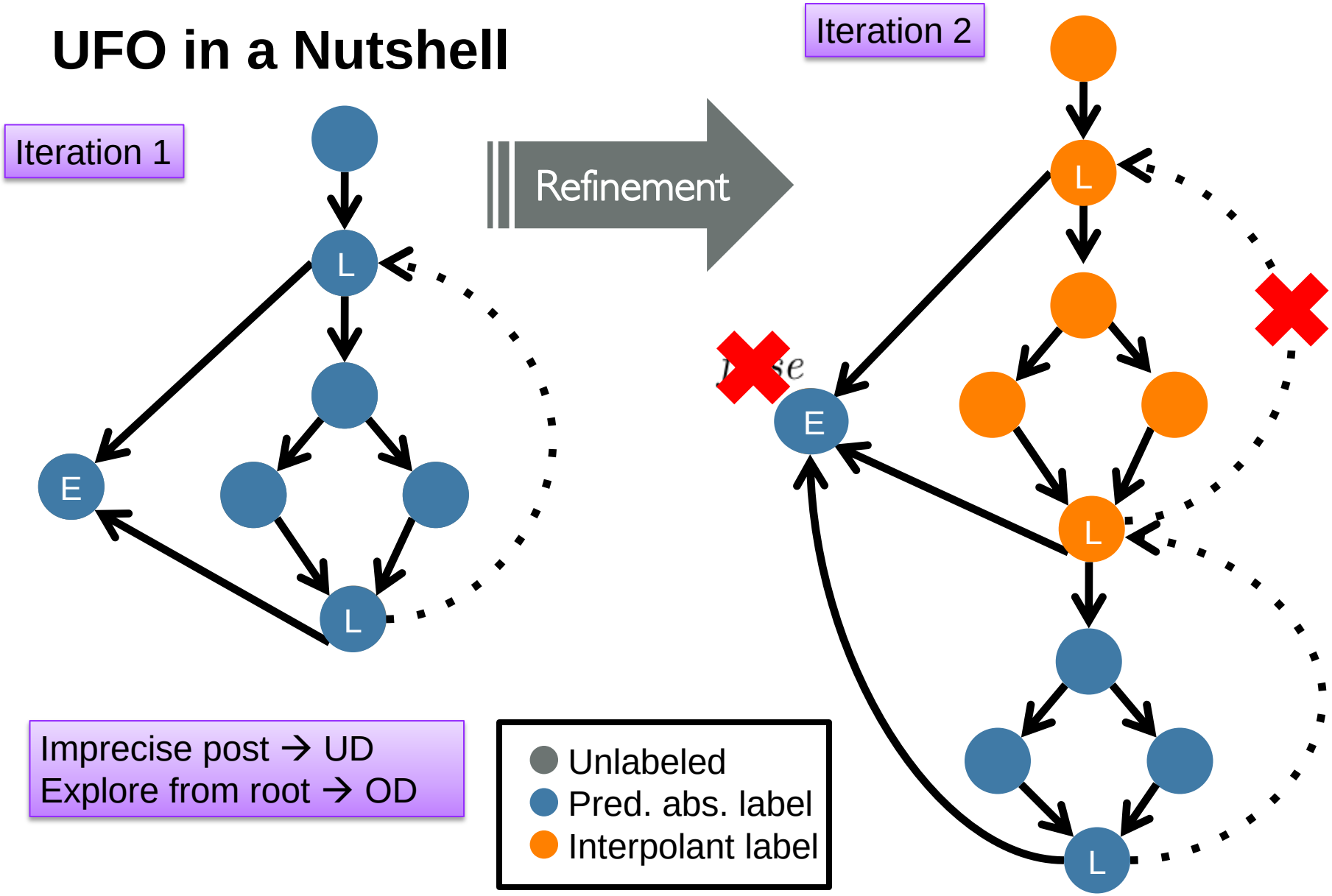
UFO in a Nutshell



UFO in a Nutshell



UFO in a Nutshell



Secret Sauce

UFO Front-End

Boxes Abstract Domain

DAG Interpolation

Parallel

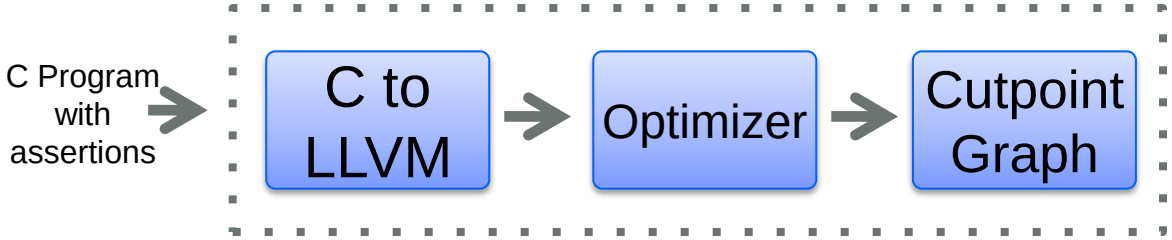
UFO Front End

In principle simple, but in practice very messy

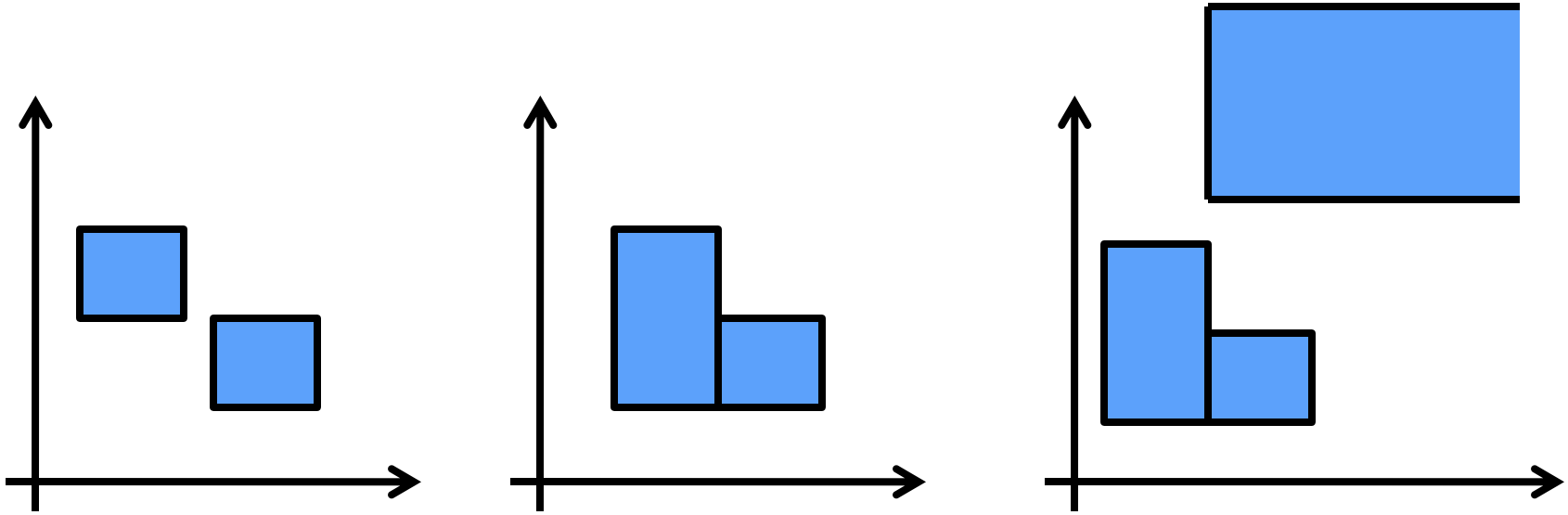
- CIL passes to normalize the code (library functions, uninitialized vars, etc.)
- `llvm-gcc` (without optimization) to compile C to LLVM bitcode
- `llvm opt` with many standard, custom, and modified optimizations
  - lower pointers, structures, unions, arrays, etc. to registers
  - constant propagation + many local optimizations
  - difficult to preserve *intended* semantics of the benchmarks
  - based on very old LLVM 2.6 (newer version of LLVM are "too smart")

Many benchmarks discharged by front-end alone

- 1,321 SAFE (out of 1,592) and 19 UNSAFE (out of 380)



Boxes Abstract Domain: Semantic View



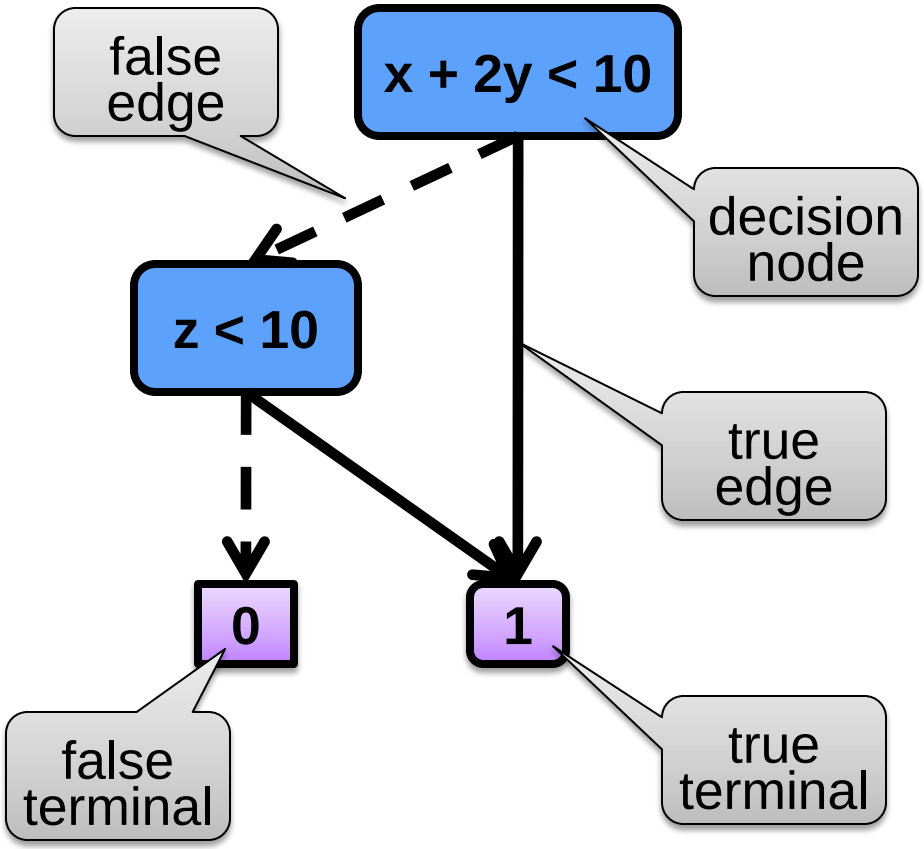
Boxes are “finite union of box values”

(alternatively)

Boxes are “Boolean formulas over interval constraints”

Linear Decision Diagrams in a Nutshell*

Linear Decision Diagram



Linear Arithmetic Formula

$(x + 2y < 10)$ **OR**
$(x + 2y \geq 10$ **AND** $z < 10)$

Compact Representation

- Sharing sub-expressions
- Local numeric reductions
- Dynamic node reordering

Operations

- Propositional (AND, OR, NOT)
- Existential Quantification

*joint work w/ Ofer Strichman

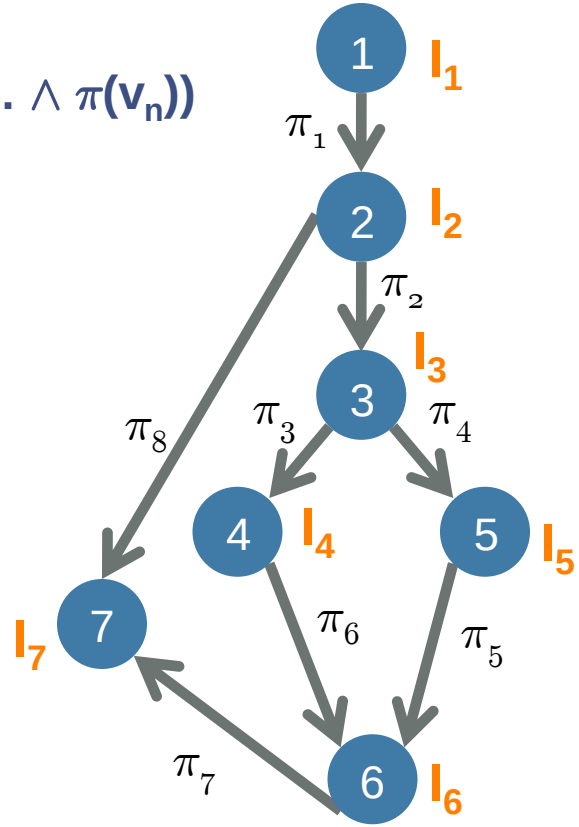
DAG Interpolants: Solving the Refinement Prob.

Given a DAG G = (V, E) and a labeling of edges π:E→Expr. A ***DAG Interpolant*** (if it exists) is a labeling I:V→Expr such that

- for any path $v_0, \ldots, v_n$, and $0 < k < n$,
  $\mathbf{I(v_k) = ITP\ (\pi(v_0) \wedge \ldots \wedge \pi\ (v_{k-1}),\quad \pi(v_k) \wedge \ldots \wedge \pi(v_n))}$
- $\forall\ \mathbf{(u, v) \in E\ .\ (I(u) \wedge \pi\ (u, v)) \Rightarrow I(v)}$

$$\mathbf{I_2 = ITP\ (\pi_1,\quad \pi_8)}$$
$$\mathbf{I_2 = ITP\ (\pi_1,\quad \pi_2 \wedge \pi_3 \wedge \pi_6 \wedge \pi_7)}$$
$$\ldots$$

$$\mathbf{(I_1 \wedge \pi_1) \Rightarrow I_2}$$
$$\mathbf{(I_2 \wedge \pi_8) \Rightarrow I_7}$$
$$\mathbf{(I_2 \wedge \pi_2) \Rightarrow I_3}$$
$$\ldots$$

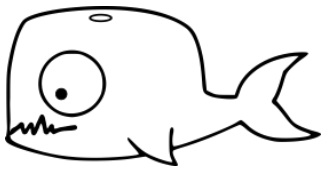


Parallel Verification Strategy

Run 7 verification strategies in parallel until a solution is found

- **cpredO3**
  - all LLVM optimizations + Cartesian Predicate Abstraction
- **bpredO3**
  - all LLVM optimizations + Boolean PA + 20s TO
- **bigwO3**
  - all LLVM optimizations + BOXES + non-aggressive widening + 10s TO
- **boxesO3**
  - all LLVM optimizations + BOXES + aggressive widening
- **boxO3**
  - all LLVM optimizations + BOX + aggressive widening + 20s TO
- **boxesO0**
  - minimal LLVM optimizations + BOXES + aggressive widening
- **boxbpredO3**
  - all LLVM opts + BOX + Boolean PA + aggressive widening + 60s TO

UFO Family



Whale [VMCAI12]

- Interpolation-based interprocedural analysis
- Interpolants as procedure summaries
- State/transition interpolation
    - a.k.a. Tree Interpolants

UFO [TACAS12]

- Refinement with *DAG interpolants*
- Tight integration of interpolation-based verification with predicate abstraction

Vinta [SAS12]

- Refinement of Abstract Interpretation (AI)
- AI-guided DAG Interpolation

Thank You!

http://bitbucket.org/arieg/ufo

Contact Information

Presenter
Arie Gurfinkel
RTSS
Telephone: +1 412-268-7788
Email: arie@cmu.edu

U.S. mail:
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web:
www.sei.cmu.edu
http://www.sei.cmu.edu/contact.cfm

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257

THE END