CPA-BAM-Slicing:

Block-Abstraction Memoization and Slicing with Region-Based Dependency Analysis [1]

Pavel Andrianov, Vadim Mutilin, Mikhail Mandrykin and Anton Vasilyev

ISP RAS

TACAS 2018, Apr 19th, 2018

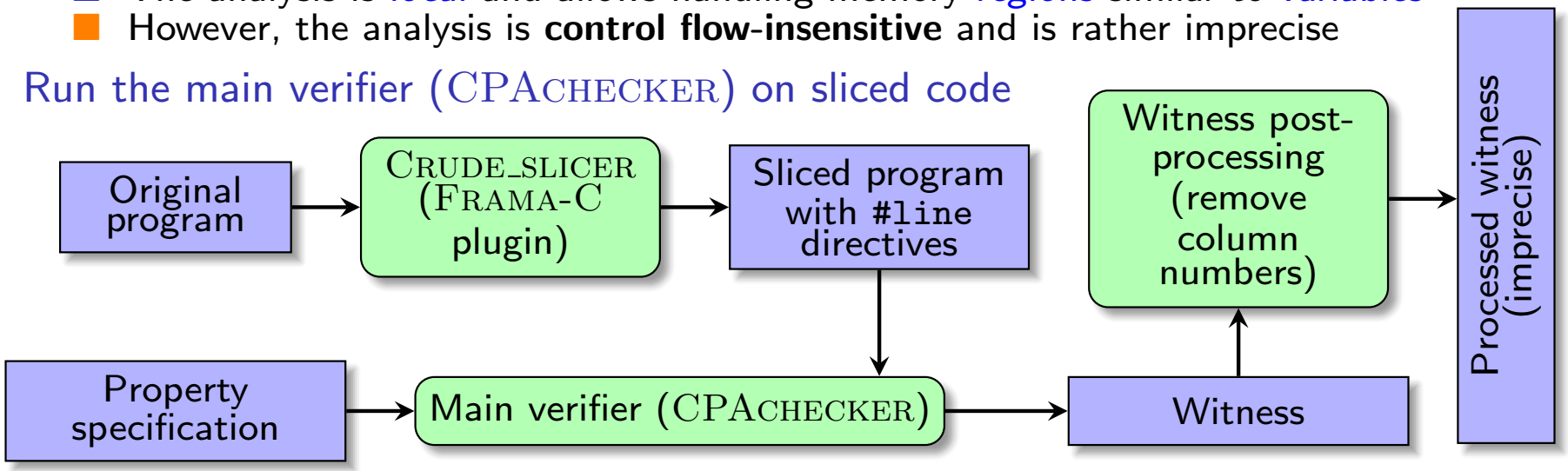
[1] The research was supported by RFBR grant 18-01-00426

Verification approach & architecture

Reachability slicing based on dataflow analysis

- In general, *transitive-closure-based slicing* is well-known
- The challenge is how to make it modular in presence of *pointer aliasing*
- We used *unification-based separation analysis* assigning *regions* to expressions
  Similar to Steensgaard's, but
  - Context sensitive – the parameter regions are *polymorphic*
  - Supports recursive types – *cycles* in region *graph* (e.g. `list->next`)
  - Partially supports *nested structures* and *pointer arithmetic*
    (`entry.head->next`, **`container_of`**`(p, entry, list_head)`)
  - Partially supports *unions* and *pointer type casts*
  - Still unsound for some type casts (e.g. pointer to integer)
- The analysis is *local* and allows handling memory *regions* similar to *variables*
- However, the analysis is **control flow-insensitive** and is rather imprecise

Run the main verifier (CPAchecker) on sliced code



Limitations

Incompleteness in presence of non-termination

- The slicer considers *only data/control flow dependencies*
- It assumes that *any program statement* **terminates**
- Target state *may be unreachable due to non-termination*, e.g.
  - Waiting for resource acquisition is an infinite loop (ignoring concurrency)
  - Special functions e.g. `__VERIFIER_assume` and `abort` semantically include infinite loops: `if (!assumption) while (true);`
- Special functions are *partially supported* via additional pairwise dependencies
- Still at least **3** wrong `FALSE` verdicts (`Systems_DeviceDriversLinux64_ReachSafety`)

Inaccurate witnesses

- *Witnesses* are produced for *sliced programs*
- They are *inaccurate* on the *original programs*
  - **None** of *violation* and only **1162** out of **2252** *correctness* witnesses were accepted in the competition

Benchmarks & results

SV-COMP `Systems_DeviceDriversLinux64_ReachSafety`

- # of benchmarks: **2734**
- Competition settings: time limit **900**s, memory limit **16**GB
- Average slicing time: **14.82**s
- Results: *with* slicing vs. *without* slicing, correct verdicts only

	TRUE verdicts			FALSE verdicts		
Config	New (+)	Lost (−)	Total	New (+)	Lost (−)	Total
BAM	151	10	2252	97	11	267
LDV	474	7	1949	27	3	282

- **Total** CPU time reduction (including timeouts):
  1.95× for BAM and **1.85**× for LDV

Future work

Summaries

- Region-based dataflow analysis can approximate *function side effects*
- If augmented with some sound symbolic execution, initial *sound function summaries* can be computed

Scope-bouned verification

- Initially function calls can be approximated by *sound summaries*
- The analysis scope can be extended based on counterexamples by expanding only relevant function calls