SL-COMP 2019

Presented by Mihaela Sighireanu

TOOLympics 2019, April 6th

Outline

SL-COMP: Competition of Solvers for Separation Logic

- Objectives:
  - promote the implementation effort on solvers for SL
  - share a benchmark of interesting problems
  - compare techniques
- History:
  - 2014: at FLOC, 6 solvers, 600 problems, 5 divisions
  - 2018: at FLOC, 11 solvers registered, 1268 problems, 11 divisions
- Achievements of the 3rd edition:
  - all 11 solvers participated
  - compliance with the common input format based on SMT-LIB 2.6
  - discussion on the scoring system
  - visibility

Input

Entailment or satisfiability problem in

- Separation Logic theory

$$\varphi \quad ::= \quad \phi \mid \mathsf{emp} \mid \mathsf{t} \mapsto \mathsf{u} \mid \varphi_1 * \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid \exists x^\sigma . \; \varphi_1(x) \mid P(x_1, \ldots, x_n)$$
$$\mid \varphi_1 \;\mathbin{-\!\!*}\; \varphi_2 \mid \neg\varphi_1 \mid$$

- with predicates defined by rules of the form

$$P(x_1, \ldots, x_n) \quad \leftarrow \quad \varphi_P(x_1, \ldots, x_n),$$

Example of Input

$$x \mapsto \texttt{node}(1, y) \ * \ y \mapsto \texttt{node}(1, z) \ * \ \texttt{ls}(z, \mathsf{nil}) \ \wedge \ z \neq x \quad \vdash \quad \texttt{ls}(x, \mathsf{nil})$$

where

$$\texttt{ls}(h, f) \ \leftarrow \ h = f \wedge \mathsf{emp}$$
$$\texttt{ls}(h, f) \ \leftarrow \ \exists x, i \, . \, h \neq f \ \wedge \ h \mapsto \texttt{node}(i, x) \ * \ \texttt{ls}(x, f)$$

Input Format

Based on SMT-LIB 2.6:

```
(theory SepLogicTyped

 :funs ((emp Bool)
        (sep Bool Bool Bool :left-assoc)
        (wand Bool Bool Bool :right-assoc)
        (par (L D) (pto L D Bool))
        (par (L) (nil L))
        )
)
```

extended with a command `declare-heap` for heap typing.

Example of Input

```
(declare-sort Loc 0)

(declare-datatype Data ((node (d Int) (next  U))))

(declare-heap (Loc Data))

(define-fun-rec ls ((h Loc) (f Loc)) Bool
    (or (and emp (= h f))
        (exists ((x Loc) (d Int))
                (and (distinct h f) (sep (pto h (node d x))
                                         (ls x f))))
    )
)
```

Fragments

- `QF`: quantifier free
- `SH`: *symbolic heap fragment*
- `ID`: user defined predicates (lists, trees, . . .)
- `LS`: restricted to list segment predicate
- `LID`: *linear* user defined predicates
- `B`: boolean combination of atoms
- `LIA`: integer `Data` and linear arithmetics constraints

Divisions

= (fragment, problem)

Division	size
qf_bsl_sat	46
qf_bsllia_sat	24
qf_shid_entl	312
qf_shid_sat	99
qf_shidlia_entl	61
qf_shidlia_sat	33
qf_shlid_entl	60
qf_shls_entl	296
qf_shls_sat	110
shid_entl	73
shidlia_entl	181

Calendar and Places

- On sl-comp@googlegroups.com and EasyChair
  - Call for solvers (and its *contact person*)
  - Call for problems
- On github.com/sl-comp/SL-COMP19
  - Benchmark set publication
  - Tools for parsing
- On StarExec: community SL
  - Space SL-COMP19
  - Two runs on special queue
  - Configuration: 10 GB and 600 seconds (LS) or 2400 seconds
- On sl-comp.github.io: results on readable format
  - Different scoring schemes

Participants

| Solver	Affiliation	Team
Asterix	TU Munich, Germany	A. Rybalchenko (MSR),
		J.A. Navarro Pérez (Google)
ComSPEN	ISCAS, China	C. Gao, Z. Wu
CVC4-SL	University of Iowa, USA	A. J. Reynolds
CYCLIST-SL	Middlesex University London, UK	N. Gorogiannis
Harrsh	TU Wien, Austria	J. Katelaan, F. Zuleger
	RWTH Aachen University, Germany	Ch. Matheja, T. Noll
S2S	Teesside University, Middlesbrough, UK	Quang Loc Le
SLEEK	NUS, Singapore	Benjamin Lee
SLIDE	FIT, Brno University of Technology, Czechia	M. Cyprian, A. Rogalewicz, T. Vojnar
	VERIMAG, Univ. of Grenoble & CNRS, France	R. Iosif
SLSAT	Middlesex University London, UK	N. Gorogiannis
Songbird	NUS, Singapore	Wei-Ngan Chin, Quang-Trung Ta,
		Thanh-Toan Nguyen, Siau-Cheng Khoo
	Stevens Institute of Technology, USA	Ton-Chanh Le
SPEN	IRIF, University of Paris & CNRS, France	C. Enea, M. Sighireanu
	FIT, Brno University of Technology, Czechia	O. Lengal, T. Vojnar

Participants by Underlying Technique

- *Automata*: Harrsh, SLIDE, SPEN
- *Model-based reduction to SMT*: Asterix, ComSPEN, CVC4-SL, SLSAT, SPEN
- *Proof search with lemmas*: S2S, SLEEK, SPEN
- *Cyclic proofs*: CYCLIST-SL, S2S, Songbird

Collected Set of Problems

Division	size	Contributor solver
qf_bsl_sat	46	CVC4
qf_bsllia_sat	24	CVC4
qf_shid_entl	312	CYCLIST, S2S, SLEEK, SLIDE, Songbird, SPEN
qf_shid_sat	99	CYCLIST, Harrsh, S2S, SLEEK
qf_shidlia_entl	61	ComSPEN, S2S
qf_shidlia_sat	33	ComSPEN, S2S
qf_shlid_entl	60	ComSPEN, SPEN
qf_shls_entl	296	Asterix, S2S, SPEN
qf_shls_sat	110	Asterix
shid_entl	73	SLEEK, Songbird
shidlia_entl	181	Songbird

Contribution to Benchmark Set
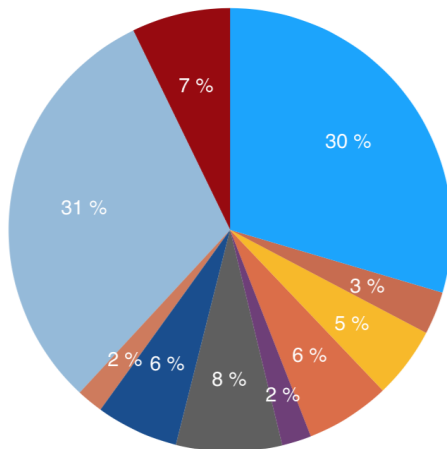
Execution on StarExec

- Solver binary running on StarExec, available for community
- Configurations
  - for LS divisions: 600 sec of timeout and 10 GB of memory
  - for other divisions: 2400 sec of timeout and 10 GB of memory
- Rounds: at least two
  - more if asked for a solver

Scoring

- Studied schemes:
  - SMT-COMP scheme: lexicographical ordering on
    - elimination if wrong results ☹
    - number of solved problems
    - total CPU time
  - *Time-reward* scheme: $1 \times solved + (-1) \times wrong + solved/solving\text{-}time$
    - penalty for any kind of wrong result
    - bonus for solving fast ☹
  - Differential penalty scheme: ✓
    $1 \times solved + (-1) \times false\text{-}positive + (-10) \times false\text{-}negative$
    - *false-positive* sound for program verification
    - CPU time is the tiebreaker
    - points for contribution to VBS (Virtually Best Solver)

Division `qf_shls_entl`

- 9 solvers, 296 problems
- Configuration: timeout 600 seconds, memory 10GB
- Podium:
    - *****: Asterix
    - ****: S2S
    - ***: SPEN
    - **: Songbird
    - *: ComSPEN
    - ☕: Cyclist-SL, Harrsh, SLEEK, SLIDE

Entry division, includes problems that reveal solver's corner cases.

Division `qf_shls_sat`

- 8 solvers, 110 problems
- Configuration: timeout 600 seconds, memory 10GB
- Podium:
    - *****: Asterix
    - ****: S2S
    - ***: ComSPEN
    - **: SLEEK
    - *: SPEN
    - ☕: Harrsh, SLSAT, Songbird

Asterix is still the best!

Division `qf_shid_entl`

- 6 solvers, 312 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
  - *****: S2S
  - ****: Songbird
  - ***: Harrsh
  - **: CyclistSL
  - *: SLEEK
  - ☕: SLIDE

Definitively a difficult division!

Division qf_shid_sat

- 5 solvers, 99 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
  - *****: S2S
  - ****: SLEEK, SLSAT
  - ***: Harrsh
  - **: Songbird

Points to SLSAT for contribution to VBS.

Division `qf_shlid_entl`

- 8 solvers, 60 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
    - *****: S2S
    - ****: Songbird
    - ***: Harrsh
    - **: Cyclist-SL
    - *: SLEEK
    - ☕: ComSPEN, SPEN, SLIDE

Fragment should be clarified to avoid false-negative results.

Division `shid_entl`

- 5 solvers, 73 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
    - *****: S2S
    - ****: Songbird
    - ***: Cyclist-SL
    - **: SLIDE
    - *:
    - ☕: SLEEK (negative score)

Divisions `qf_bsl_sat` and `qf_bsllia_sat`

- 1 solver for 46 rest. 24 problems
- Question: what to do with magic wand?
- Podium:
  - *****: CVC4-SL

Need for solvers to challenge CVC4!

Division `qf_shidlia_entl`

- 4 solvers, 61 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
    - *****: S2S
    - ****: Songbird
    - ***: ComSPEN
    - **:
    - *:
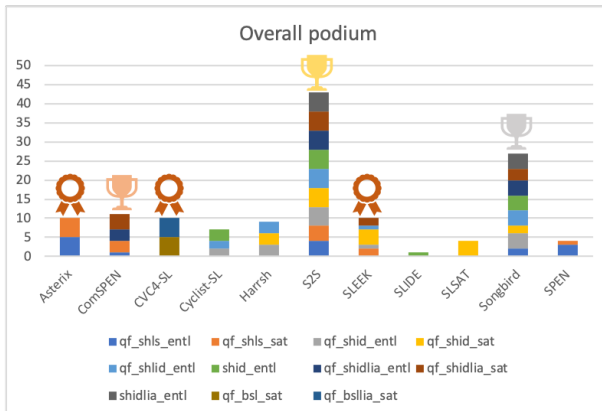   - ☕: SLEEK (negative score)

Division `qf_shidlia_sat`

- 4 solvers, 33 problems
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
    - *****: S2S
    - ****: ComSPEN
    - ***: Songbird
    - **: SLEEK

Division `shidlia_entl`

- 3 solvers, 181 problems
- Execution timeouts shall be $>=$ 2400 sec
- Configuration: timeout 2400 seconds, memory 10GB
- Podium:
    - *****: S2S
    - ****: Songbird
    - ☕: SLEEK (negative score)

Overview of Results

Conclusion and Future

Successful edition:

- consolidate newcomers and old fellows still competitive
- clean input and tools supporting it
    - more solvers are aligned

Future:

- clean existing benchmark set based on analysers of logic fragments
- add more problems issued from verification tools
- provide a witness for solving
- next edition: at ADSL 2020? TOOLympics? FLOC 2022!