



Report on the 2023 edition

<https://chc-comp.github.io/>

Emanuele De Angelis, IASI-CNR, Italy

Hari Govind V K, Univ. of Waterloo, Canada

HCVS, 23 April 2023, Paris, France



Outline

- Tracks
- Benchmarks
- Teams & Solvers
- Results
- Discussion



Tracks



Tracks

- Linear Integer Arithmetic, Linear clauses (LIA-Lin)
- LIA, Nonlinear clauses (LIA-Nonlin)
- LIA and Arrays, Linear clauses (LIA-Lin-Array)
- LIA and Arrays, Nonlinear clauses (LIA-Nonlin-Array)
- ~~Linear Real Arithmetic, Transition Systems (LRA-TS)~~
- ~~LRA-TS parallel, Transition Systems (LRA-TS-par)~~
- ~~Algebraic Data Types, Nonlinear clauses (ADT-Nonlin)~~
- LIA, Arrays and non-recursive ADT, Nonlinear clauses (LIA-Nonlin-Arrays-nonrecADT)
- **ADT and LIA, Nonlinear clauses (ADT-LIA-nonlin)**

New in 2023

Benchmarks

Inventory, Processing & Selection process



Repositories

github <https://github.com/chc-comp>

StarExec <https://www.starexec.org/starexec/secure/explore/spaces.jsp?id=73700>

New Benchmarks on Algebraic Data Types + LIA

- **ADTRem** (Thanks to Fabio Fioravanti)
Source: ADTRem tool benchmark set; taken from CLAM, HipSpec, IsaPlanner, and Leon
<https://github.com/chc-comp/ADTRem/>
- **TIP-ADT-LIA** (Thanks to Yurii Kostyukov)
Source: "Tons of Inductive Problems" (<https://github.com/tip-org/benchmarks>)
<https://github.com/ndreuu/TIP-no-NAT/releases/tag/chc-comp-23>
- **Rust-horn**
Source: RustHorn tool benchmark set
<https://github.com/hopv/rust-horn/tree/master/toplas2021/benchmarks/rust-horn>



Benchmark processing

1. Formatted according to CHC-COMP format (<https://chc-comp.github.io/format.html>) using `format.py` (<https://github.com/chc-comp/chc-tools>)
2. Categorized by background theory according to the CHC-COMP tracks using `check[-TRACK]` (<https://github.com/chc-comp/chc-tools>)
3. Removed duplicated benchmarks

Step 1 & Step 2:
**New formatter & checker
supporting ADTs!**



Benchmarks inventory

(total/unique
#benchmarks)

Repository	LIA- lin	LIA- nonlin	LIA- lin- Arrays	LIA- nonlin- Arrays	LIA- nonlin- Arrays- nonrecADT	ADT- LIA- nonlin
adtrem (<i>new</i>)						251/247
aeval	54/54					
aeval-unsafe	54/54					
chc-comp19			290/290			
eldarica-misc	149/136	69/66				
extra-small-lia	55/55					
hcai	101/87	133/131	39/39	25/25		
hopv	49/48	68/67				
jayhorn	75/73	7325/7224				
kind2		851/736				
ldv-ant-med			10/10	342/342		
ldv-arrays			3/2	821/546		
llreve	66/66	59/57	31/31			
quic3			43/43			
rust-horn (<i>new</i>)	11/11	6/6				56/56
seahorn	3379/2812	68/66				
solidity					2200/2174	
sv-comp	3150/2930	1643/1169	79/73	856/780		
synth/nay-horn		119/114				
synth/semgus				5371/4839		
tip-adt-lia (<i>new</i>)						320/320
tricera	405/405	4/4				
tricera/adt-arrays					156/156	
ultimate		8/8		23/23		
vmt	906/803					
total	8454/7534	10353/9648	495/488	7438/6555	2356/2330	627/623

Benchmark selection

The "hardness" of the benchmarks is determined by using the results of the two top solvers from 2022:

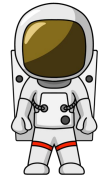
- A** - rated benchmarks: *both solvers can solve it*
- B** - rated benchmarks: *one solver can solve it*
- C** - rated benchmarks: *both solvers timed out*

Time out 30s, for both solvers

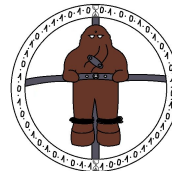
Selection process applied to tracks (without ADTs) where we have too many benchmarks:



Spacer



Spacer



Golem



ELDARICA

LIA- $\{\text{lin}, \text{nonlin}\}$

LIA-nonlin-Arrays



Benchmark selection

(Cont'd)

1. Run the two top solvers to get the ratings **A**, **B**, and **C**. This yields four sets:
 - A**-rated benchmarks
 - B**-rated benchmarks solved by Spacer only
 - B**-rated benchmarks solved by Golem | Eldarica only
 - C**-rated benchmarks
2. For each repo, we choose a number Nr of benchmarks to randomly select:
 - up to $0.2 \times Nr$ **A**-rated benchmarks
 - up to $0.4 \times Nr$ **B**-rated benchmarks equally distributed between Spacer & Golem | Eldarica
 - up to $0.4 \times Nr$ **C**-rated benchmarks

If any repo contains fewer benchmarks than required, take the rest from the next higher rating class.

Benchmark selection

(tracks wit ADTs)

The "hardness" of the benchmarks with Algebraic Data Types is determined by using the results of the top solver only (*)

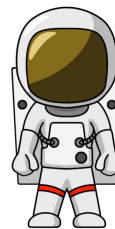
B - rated benchmarks: Spacer can *solve it*

C - rated benchmarks: Spacer *timed out* (30s)

(*) Due to limited number of CHC-COMP 2022 solvers that are able to reason on ADTs/have restrictions on the SMT-LIB syntax for ADTs.

For each repo w/ADTs, we choose a number Nr of benchmarks to randomly select

- up to $0.4 \times Nr$ **B**-rated benchmarks
- up to $0.6 \times Nr$ **C**-rated benchmarks



Spacer

Hardness statistics

Repository	LIA-nonlin-Arrays-nonrecADT		ADT-LIA-nonlin	
	#B ₁	#C	#B ₁	#C
adtrem			86	161
rust-horn			43	13
solidity	2109	65		
tip-adt-lia			39	281
tricera/adt-arrays	65	91		
total	2174	156	168	455

Repository	LIA-lin				LIA-nonlin				LIA-nonlin-Arrays			
	#A	#B ₁	#B ₂	#C	#A	#B ₁	#B ₂	#C	#A	#B ₁	#B ₂	#C
aeval	12	9	4	29								
aeval-unsafe	17	0	12	25								
eldarica-misc	120	5	9	2	39	13	0	14				
extra-small-lia	30	13	8	4								
hcai	82	1	3	1	123	0	5	3	17	3	0	5
hopv	48	0	0	0	57	3	5	2				
jayhorn	73	0	0	0	3712	2275	1	1236				
kind2					650	70	0	16				
ldv-ant-med									0	128	0	214
ldv-arrays									7	195	0	344
llreve	61	0	5	0	48	4	2	3				
rust-horn	10	1	0	0	5	0	0	1				
seahorn	2089	65	69	589	60	1	2	3				
sv-comp	2854	1	74	1	1117	40	4	8	310	330	7	133
synth/nay-horn					70	20	4	20				
synth/semgus									737	2254	4	1844
tricera/svcomp20	43	7	4	351	4	0	0	0				
ultimate					0	1	0	7	0	0	0	23
vmt	711	31	7	54								
total	6150	133	195	1056	5885	2427	23	1313	1071	2910	11	2563



Competition benchmarks

(to be selected/**selected**)

Repository	LIA-lin	LIA-nonlin	LIA-nonlin-Arrays	LIA-nonlin-Arrays-nonrecADT	ADT-LIA-nonlin
adtrem					125/125
aeval	30/30				
aeval-unsafe	30/30				
eldarica-misc	45/25	30/26			
extra-small-lia	30/22				
hcai	45/14	60/20	15/11		
hopv	30/6	30/16			
jayhorn	30/6	180/180			
kind2		90/52			
ldv-ant-med			60/60		
ldv-arrays			90/90		
llreve	30/11	45/18			
rust-horn					28/18
seahorn	90/90	45/15			
solidity				312/127	
sv-comp	90/38	90/48	135/135		
synth/nay-horn		60/48			
synth/semgus			135/135		
tip-adt-lia					160/160
tricera/svcomp20	60/60	3/0			
tricera/adt-arrays				156/122	
ultimate		6/5	15/15		
vmt	90/90				
total	600/422	639/428	450/446	468/249	313/303


Solvers (6 competing + 1 hors concours)

	LIA-lin	LIA-nonlin	LIA-lin-Arrays	LIA-nonlin-Arrays	LIA-nonlin-Arrays-nonrecADT	ADT-LIA-nonlin
Eldarica	Yes	Yes	Yes	Yes	Yes	Yes
Golem	Yes	Yes	No	No	No	No
LoAT	Yes	No	No	No	No	No
Theta	Yes	Yes	Yes	Yes	No	No
Ultimate TreeAutomizer	Yes	Yes	Yes	Yes	No	No
Ultimate Unihorn	Yes	Yes	Yes	Yes	No	No
Spacer (Hors Concours)	Yes	Yes	Yes	Yes	Yes	Yes



Competition Runs

Resources

- Timeout: **1800s** (CPU time, wall-clock time)
- Memory limit: **64GB**
- Two jobs per  StarExec node, two cores for each job
NEW nodes! Specs: Intel(R) Xeon(R) Gold 6334 CPU @ 3.60GHz

Competition results



LIA- lin

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Spacer	422	265	199	66	157	124	25	274397	138310	2675	43
Golem	422	229	148	81	193	193	0	368980	129633	1042	8
Eldarica	422	219	160	59	203	203	0	385851	112832	6644	23
Theta	422	170	122	48	252	141	3	426006	370425	14394	0
Ultimate Unihorn	422	103	72	31	319	244	0	449683	384389	15697	0
Ultimate TreeAutomizer	422	81	50	31	341	297	0	537858	517349	15396	0
LoAT	422	50	0	50	372	151	0	287878	287841	275	4

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



LIA- nonlin

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Spacer	428	384	235	149	44	39	0	90842	50781	855	38
Eldarica	428	330	185	145	98	98	0	218944	79522	6675	9
Golem	428	310	178	132	118	118	0	248569	248578	281	3
Ultimate Unihorn	428	121	72	49	307	242	0	470768	389915	16056	0
Theta	428	38	8	30	390	380	0	687374	666145	14575	0
Ultimate TreeAutomizer	428	34	5	29	394	311	0	569895	531158	15636	0

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



LIA- lin- Arrays

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Spacer	488	281	212	69	207	199	0	359439	187454	606	81
Eldarica	488	220	150	70	268	264	0	478284	166185	7589	15
Theta	488	184	134	50	304	32	0	285884	271624	16085	0
Ultimate Unihorn	488	164	122	42	324	126	0	242113	206799	18029	1
Ultimate TreeAutomizer	488	131	96	35	357	131	0	239591	229783	17810	0

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



LIA- nonlin- Arrays

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Spacer	446	258	148	110	188	142	0	290925	156914	1138	75
Eldarica	446	206	122	84	240	240	0	454921	184851	6914	26
Ultimate Unihorn	446	96	37	59	350	120	0	234519	199416	16427	0
Theta	446	85	45	40	361	281	0	588095	569760	15182	4
Ultimate TreeAutomizer	446	56	6	50	390	149	0	276025	250747	16284	0

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



LIA-nonlin-Arrays-nonrecADT

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Eldarica	249	176	85	91	73	57	0	114521	42212	3845	57
Spacer	249	120	59	61	129	107	0	195321	107046	190	1

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



ADT-LIA-nonlin

solver	cnt	ok	sat	uns	fld	to	mo	time	real	space	uniq
Eldarica	303	58	22	36	245	239	0	433561	150012	4754	30
Spacer	303	30	3	27	273	237	0	440259	290358	411	2

cnt number of benchmarks
ok number of solved benchmarks
sat number of SAT solved
uns number of UNSAT solved
fld number of failed unsolved
to number of timeouts

mo number of memory outs
time sum of total cpu time in seconds
real sum of total wall-clock time in seconds
space sum of memory used in MB
uniq number of unique benchmarks solved



Results

	LIA-lin	LIA-nonlin	LIA-lin- Arrays	LIA-nonlin- Arrays	LIA-nonlin- Arrays- nonrecADT	ADT-LIA- nonlin
1st	Golem	Eldarica	Eldarica	Eldarica	Eldarica	Eldarica
2nd	Eldarica	Golem	Theta	Ultimate Unihorn		
3rd	Theta	Ultimate Unihorn	Ultimate Unihorn	Theta		

+ Spacer for many unofficial 1st places

Big Thanks to





Discussion

- **Result validation**
 - rules to handle cases when solvers disagree on result
 - models/counterexamples
- **Benchmarks**
 - new benchmark set for LIA-`{lin, nonlin}`, LIA-`{lin, nonlin}`-Arrays
 - `set-info :status`
- **Tracks**
 - Tracks w/ADT, new solvers are welcome!
 - Parallel tracks for portfolio solvers
 - Adding a more general LRA track
- **Organizers of the next edition**