Scalable Proof Producing Multi-Threaded SAT Solving with Gimsatul through Sharing instead of Copying Clauses

Mathias Fleury and Armin Biere

Albert-Ludwig-Universität Freiburg, Germany
fleury@cs.uni-freiburg.de biere@cs.uni-freiburg.de

13th Pragmatics of SAT international workshop

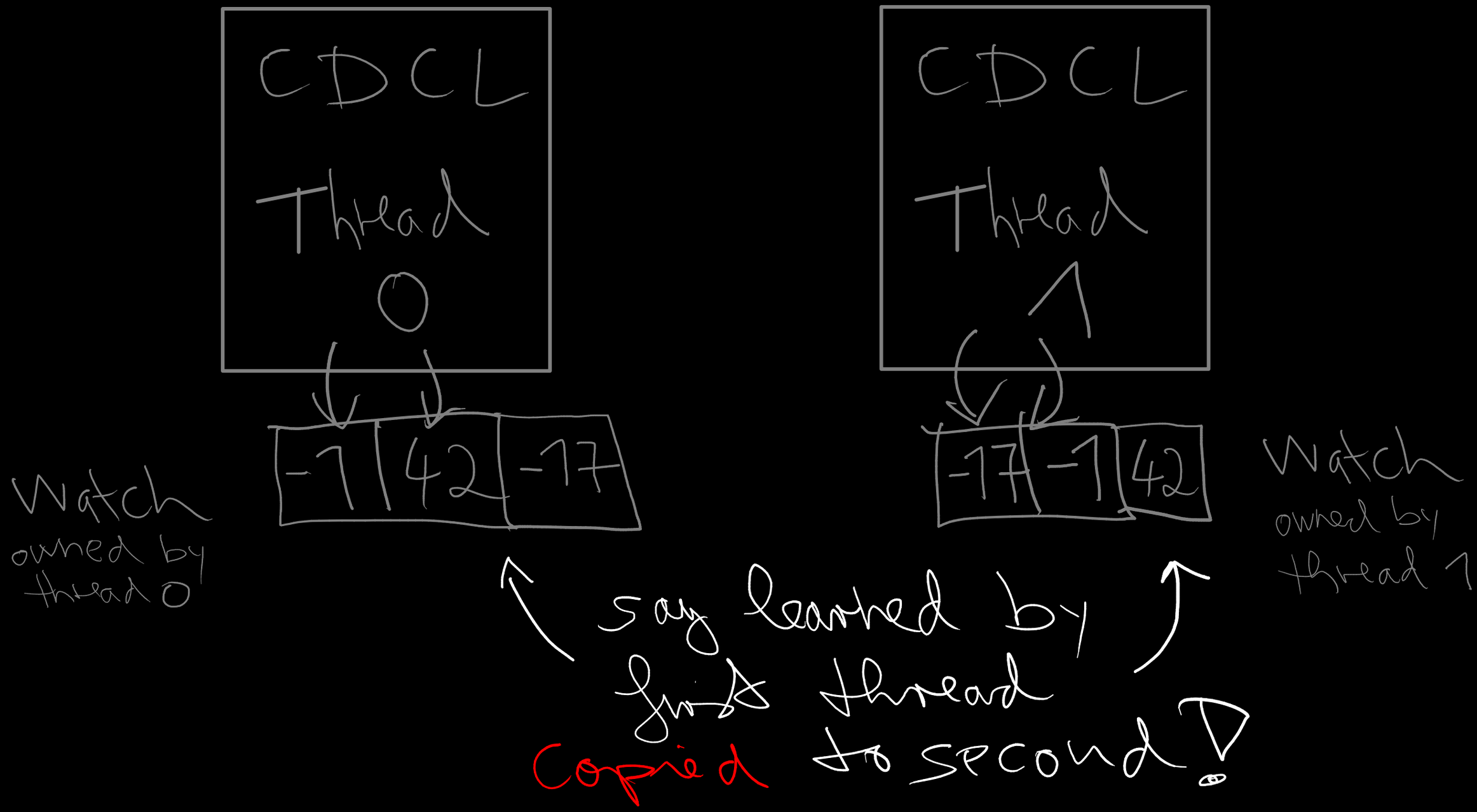
a workshop of the 25th International Conference on Theory and Applications of Satisfiability Testing

part of the Federated Logic Conference (FLoC) 2022

August 1, 2022, Haifa, Israel

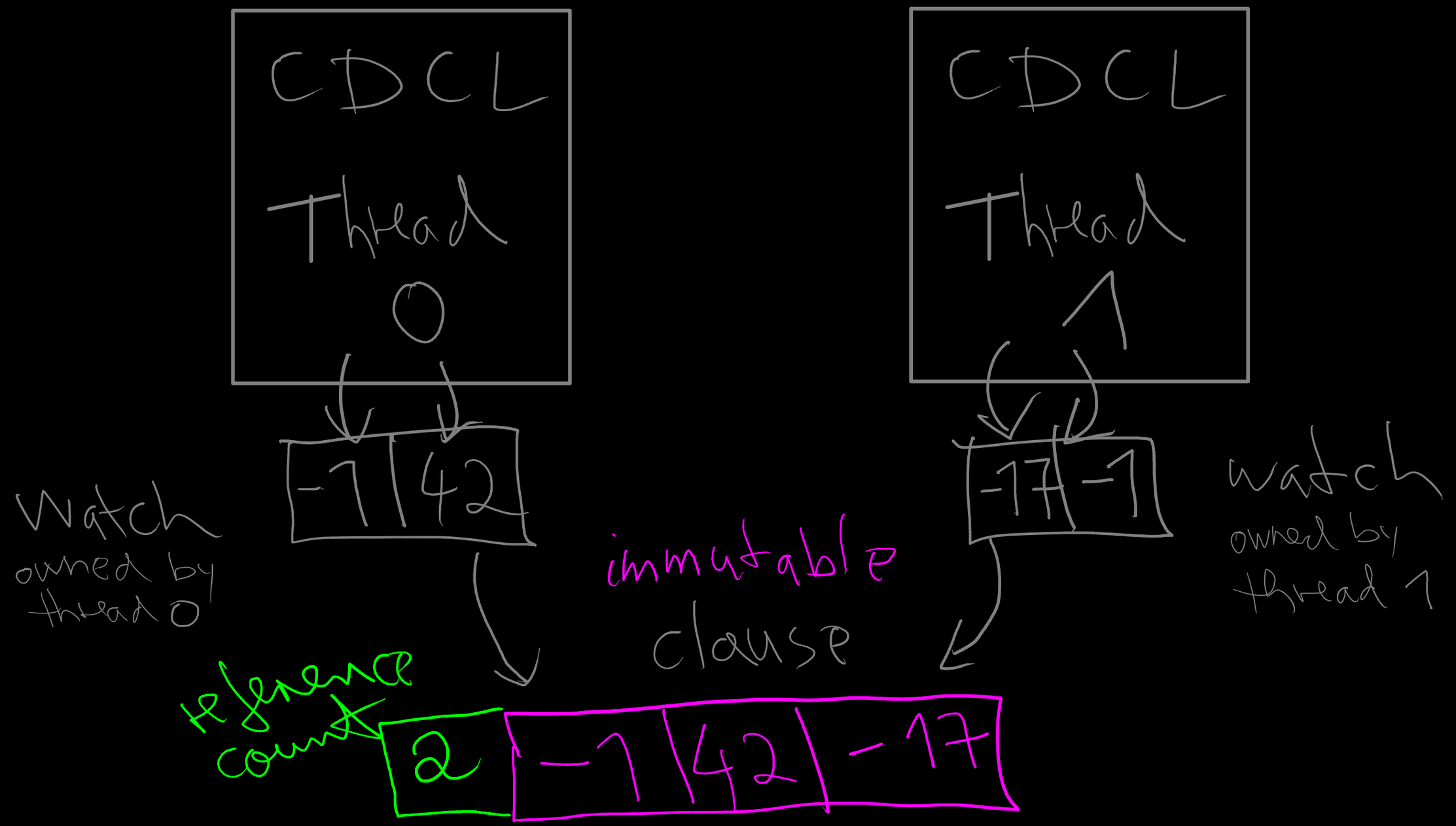
To Share or to Copy?

Many SAT



To Share or to Copy?

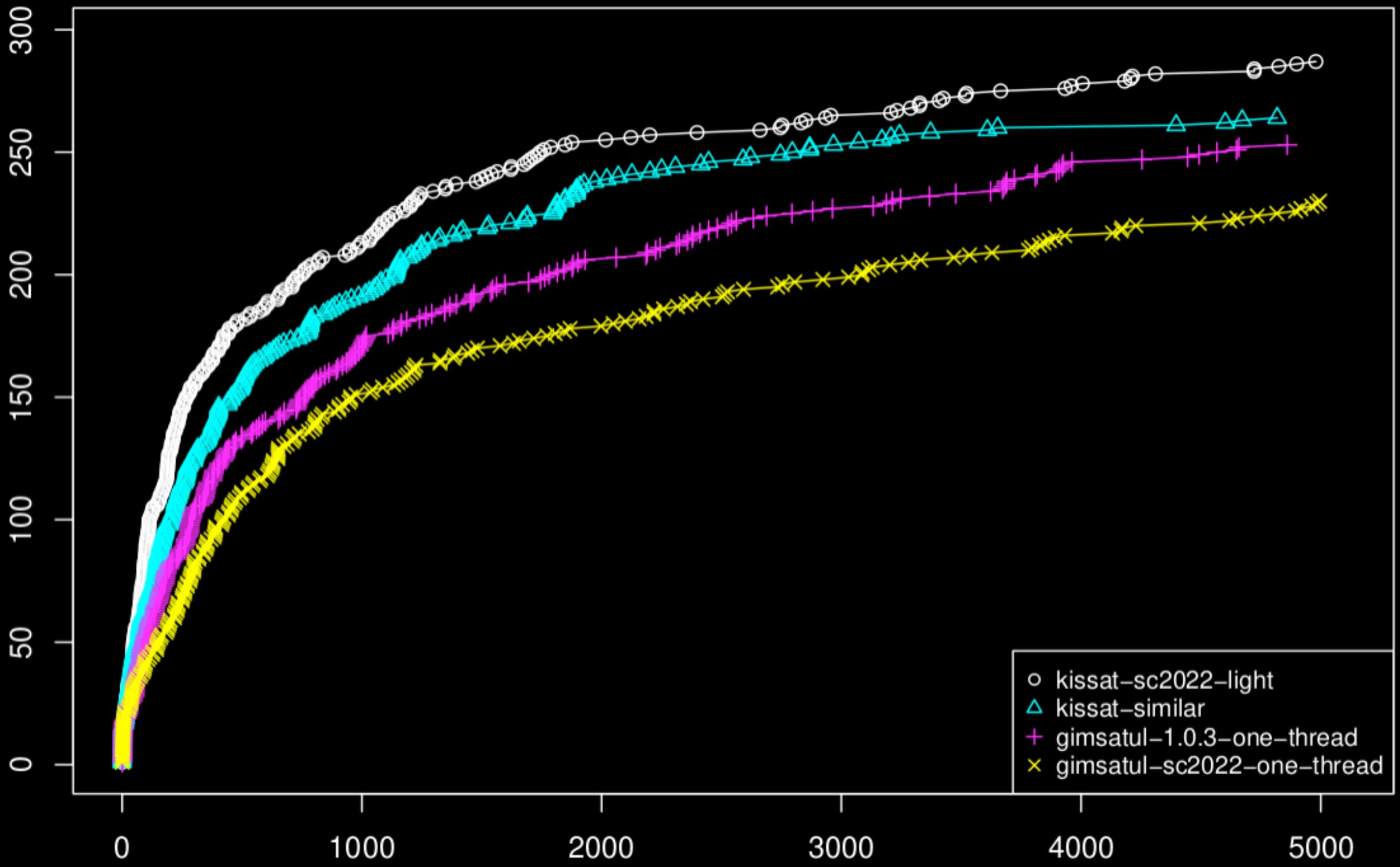
Gimsatul



More Gimsatul Features

1. virtual learned binary clauses
2. completely shared original binary clauses
   including their watcher stacks!
3. export all low glue clauses to all threads immediately
4. import ONE clause before decision all units eagerly

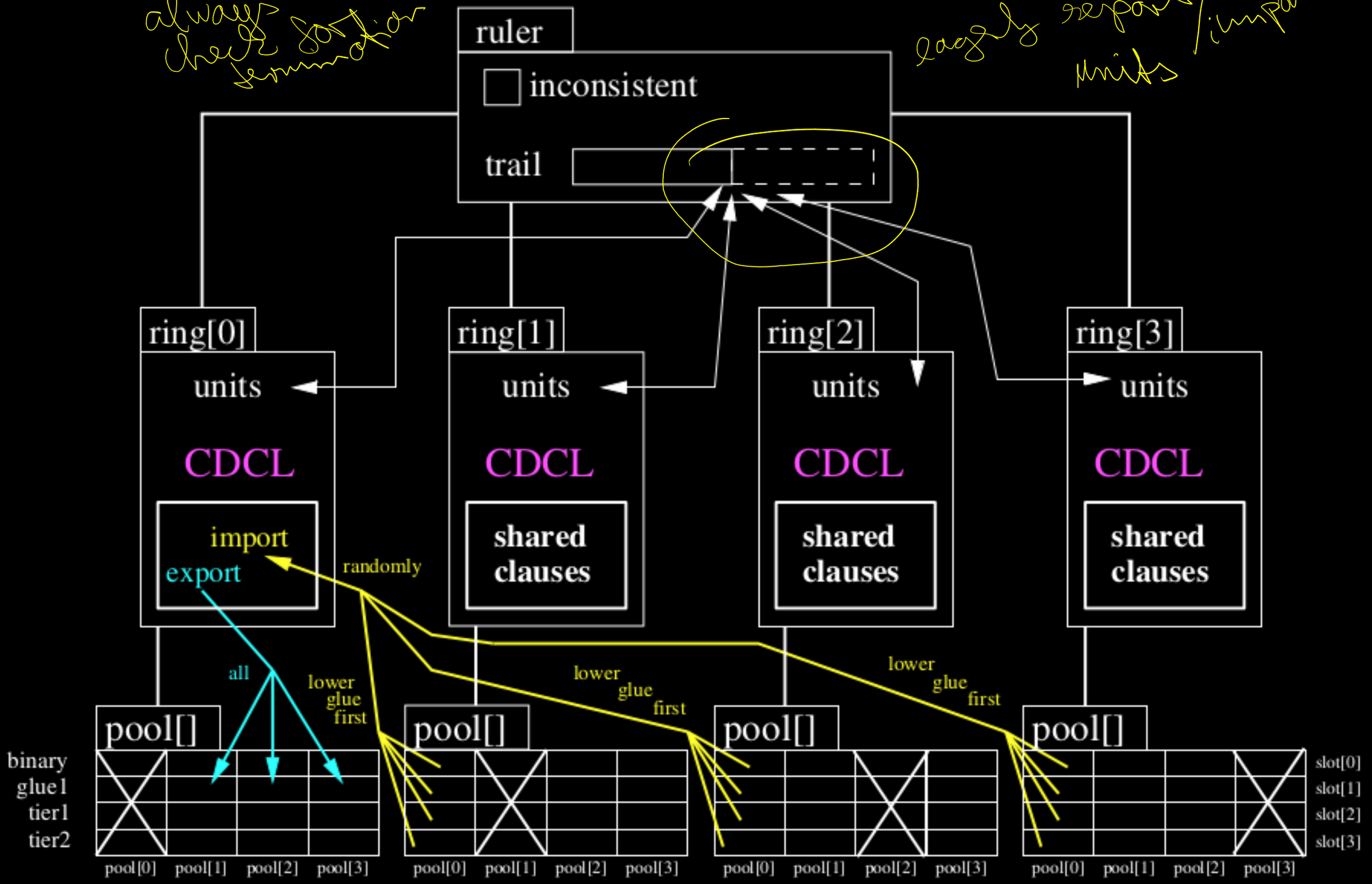
GIMSATUL Vs KISSAT



Architecture

always check termination

eagerly export units / import



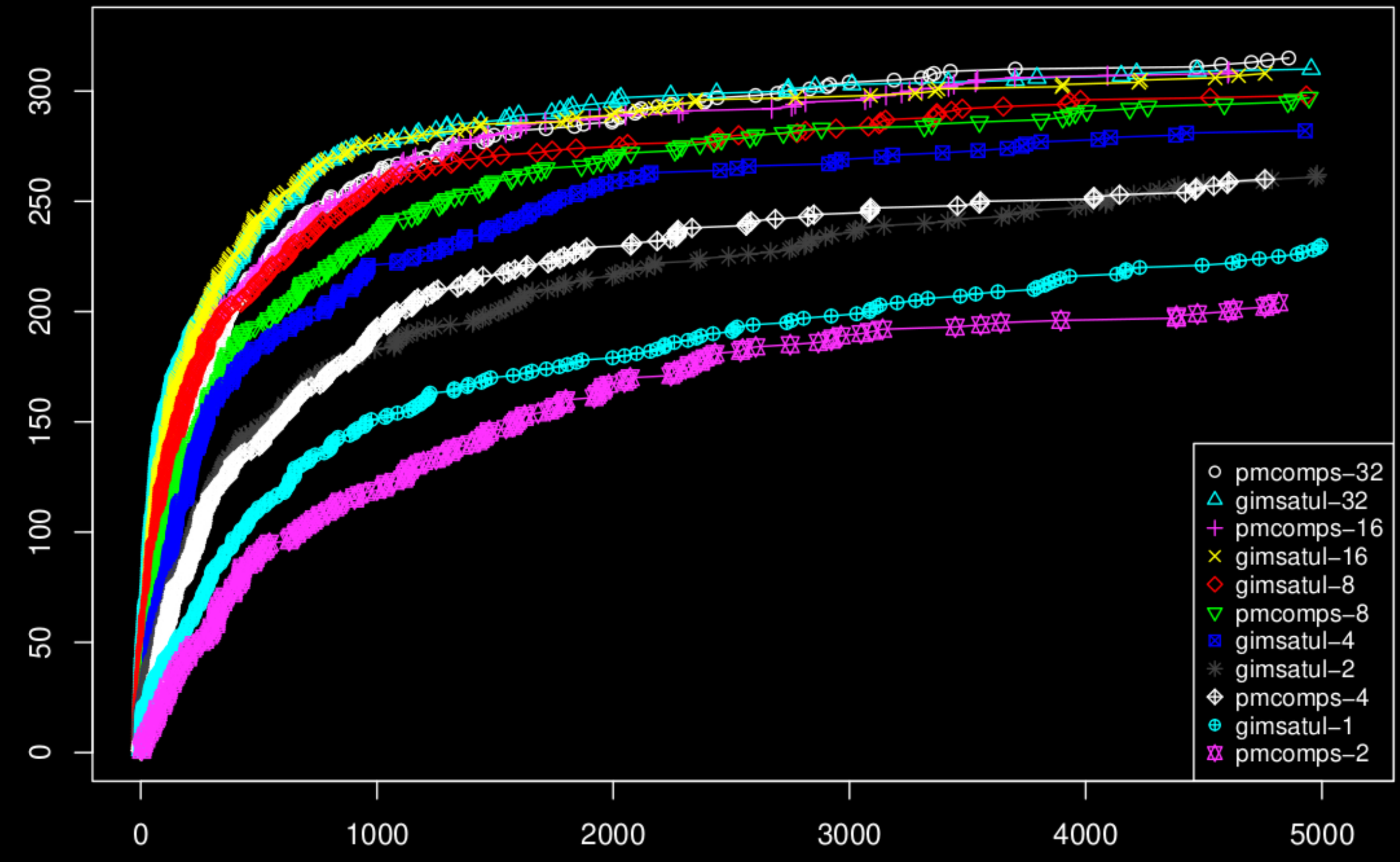
Results SAT Comp 2021 (Main Track)

#threads

	solved	sat	uns	elapsed time (s)	PAR-2 (10^3)	space
GIMSATUL-32	310	**151**	159	3 928 732	1 031	1 578 816
P-MCOMSPS-32 *	**315**	141	**174**	5 804 244	1 037	3 038 965
GIMSATUL-16	308	149	159	2 069 523	1 055	843 444
P-MCOMSPS-16	309	136	173	2 607 129	1 076	1 495 201
GIMSATUL-64	304	150	154	4 768 809	1 119	2 948 359
GIMSATUL-8	298	144	154	1 223 199	1 178	460 013
P-MCOMSPS-8	297	131	166	1 584 009	1 229	723 740
GIMSATUL-4	282	138	144	744 566	1 371	263 227
P-MCOMSPS-4	260	119	141	854 403	1 614	317 691
GIMSATUL-2	262	130	132	499 420	1 633	167 771
GIMSATUL-1	230	117	113	263 851	1 963	80 947
P-MCOMSPS-2	204	87	117	679 142	2 187	187 676

* winner parallel Track 2021

Results SAT Comp 2021
(Main Track)



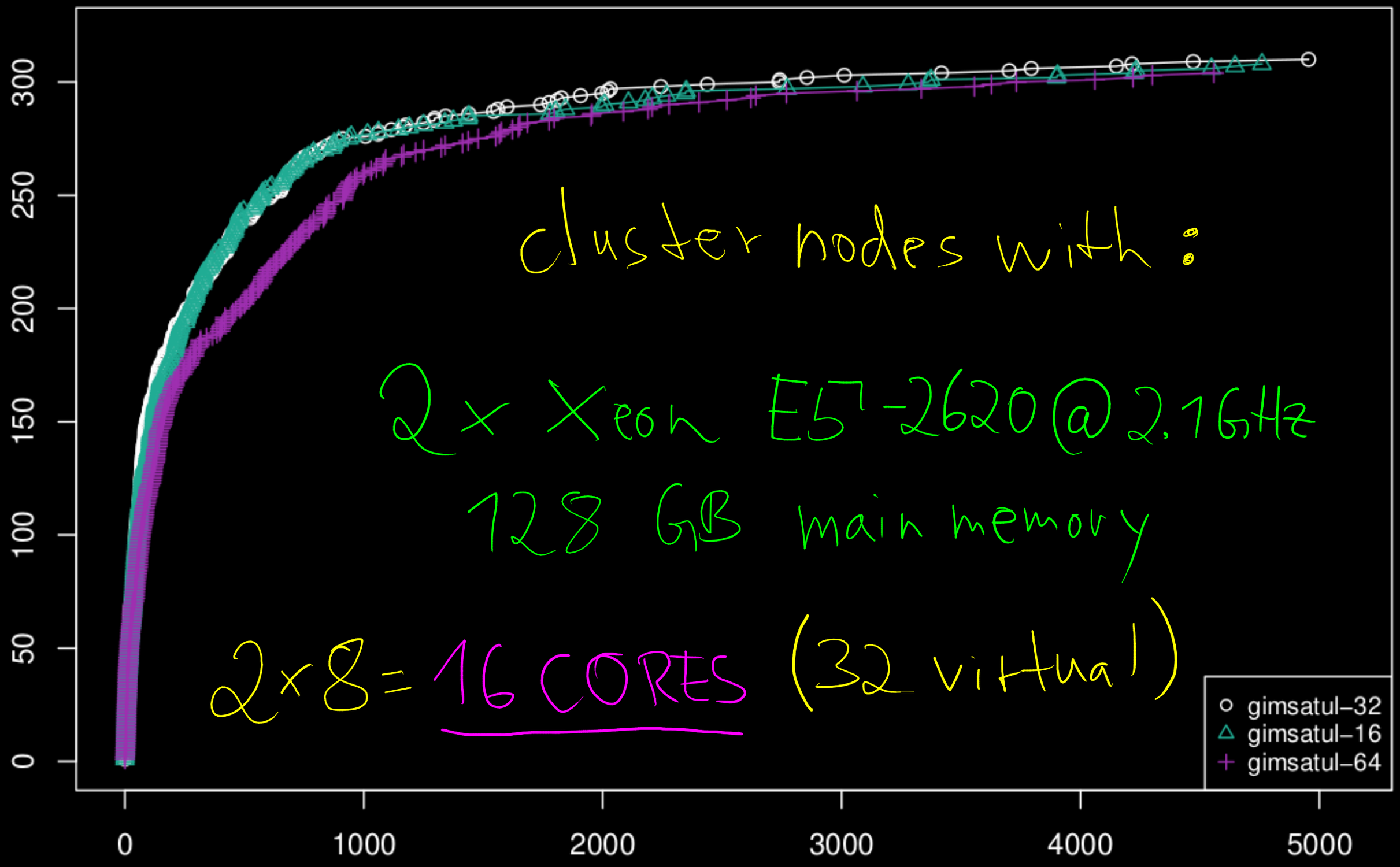
More Threads Than REAL Cores

cluster nodes with:

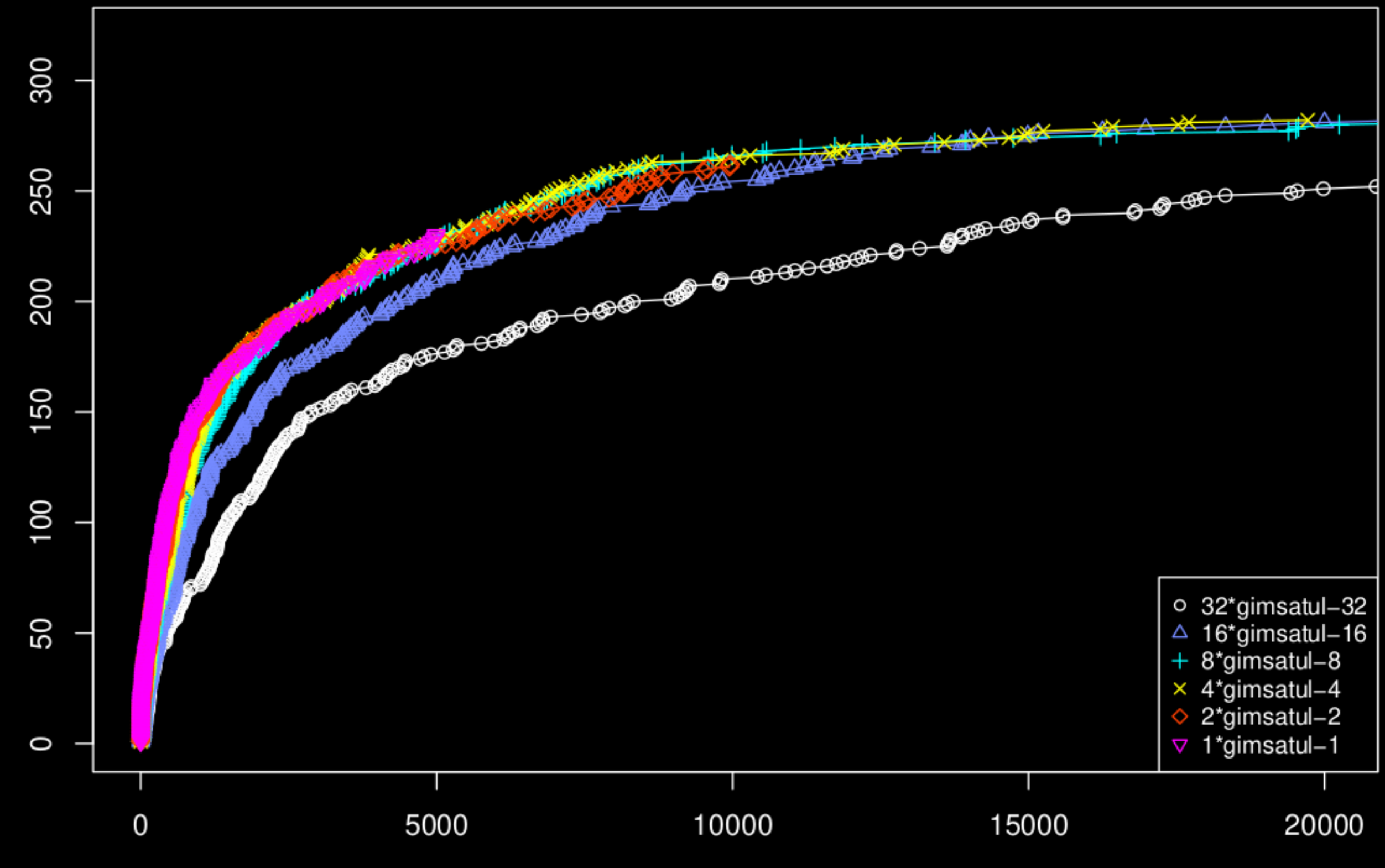
2 x Xeon E5-2620 @ 2.1GHz

128 GB main memory

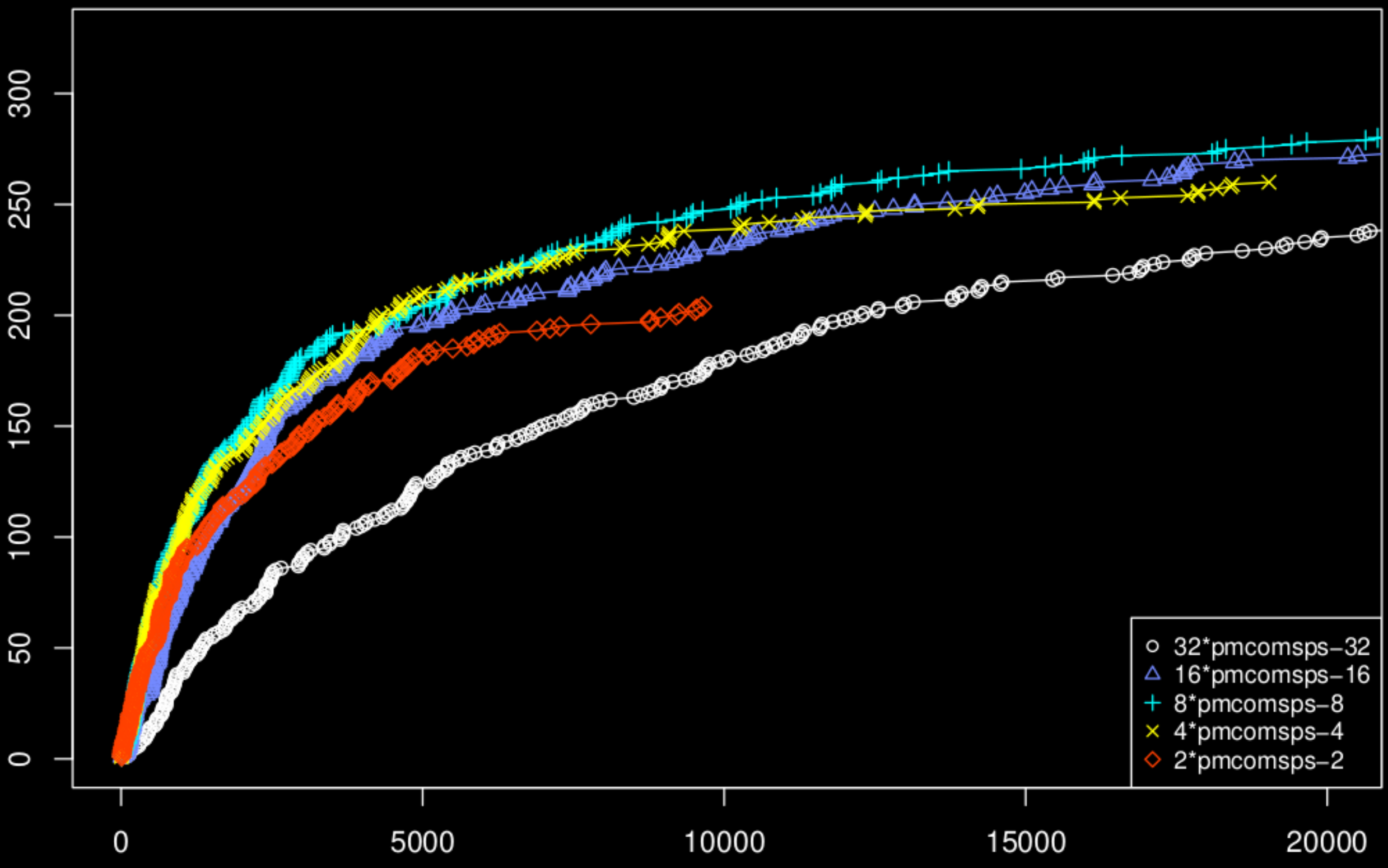
2x8 = 16 CORES (32 virtual)



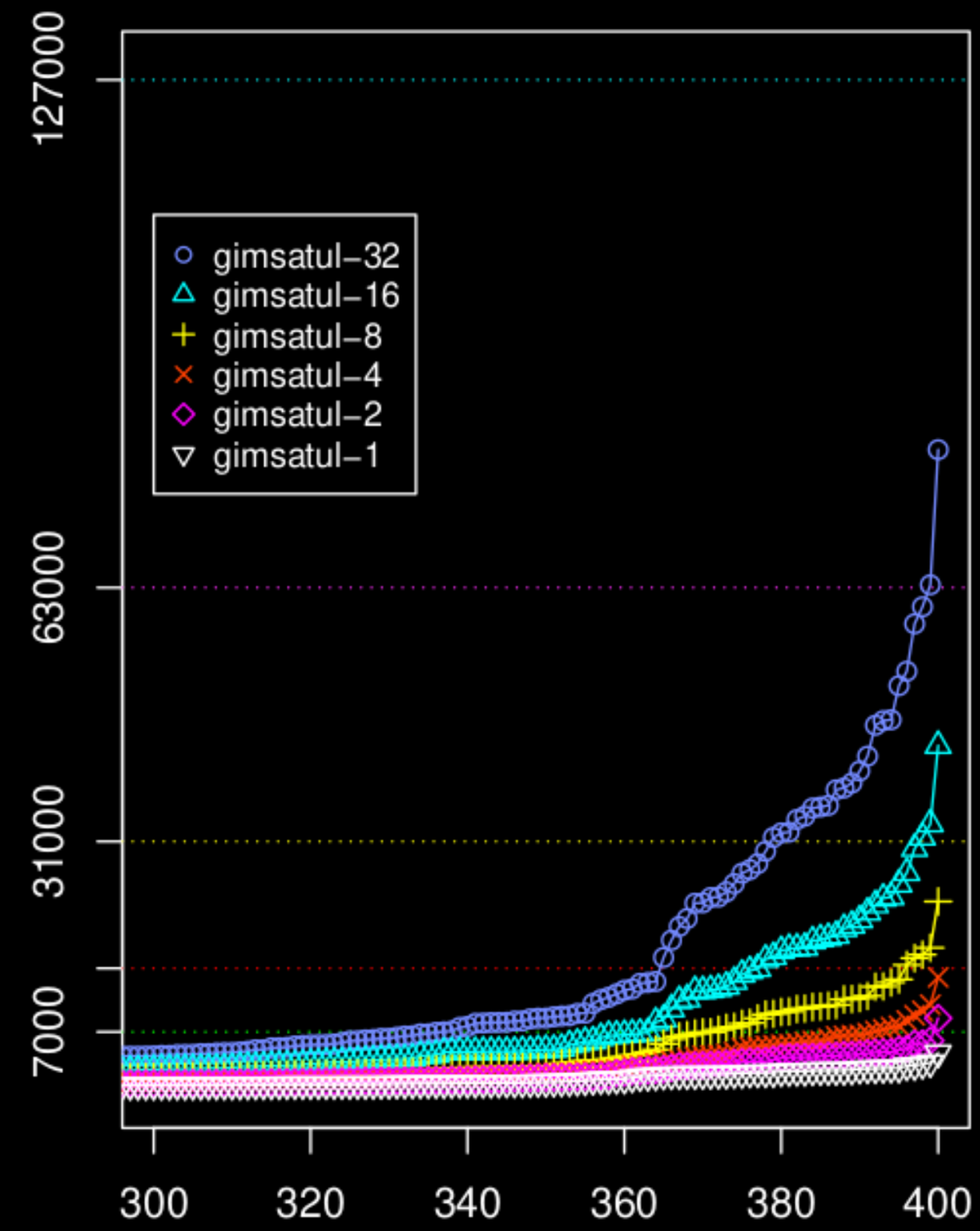
Wall-Clock Scalability

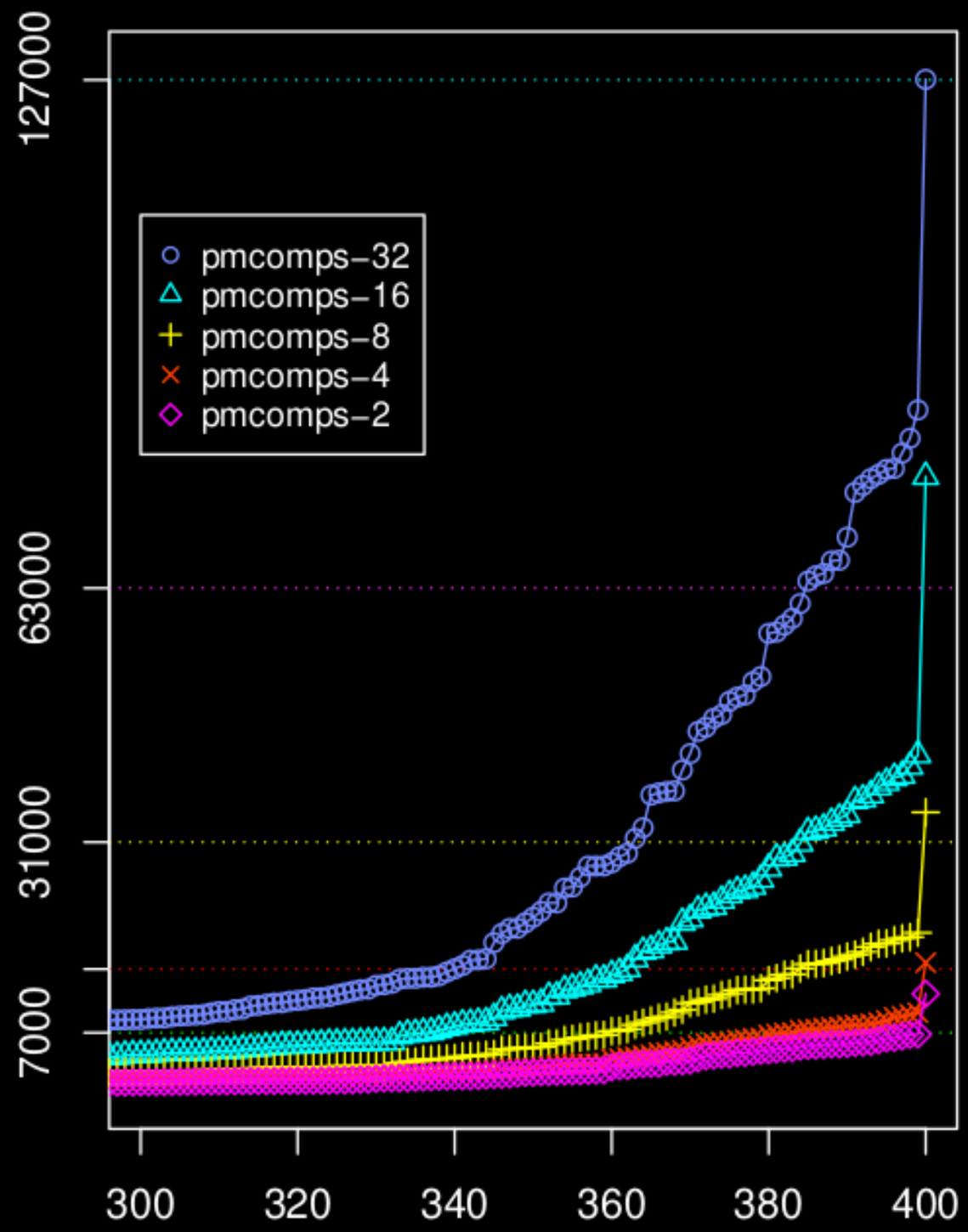


Wall-Clock Scalability

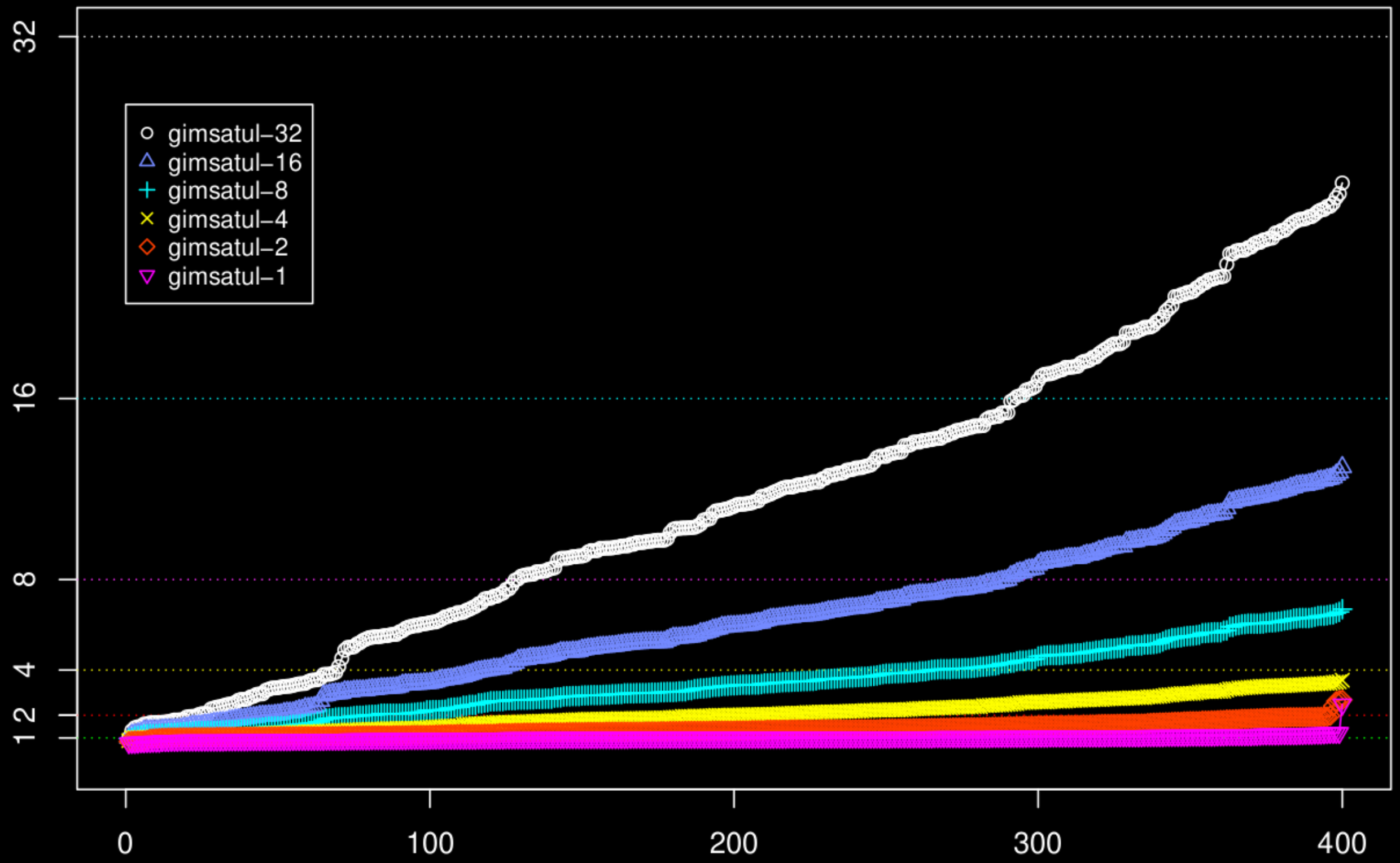


Memory Usage



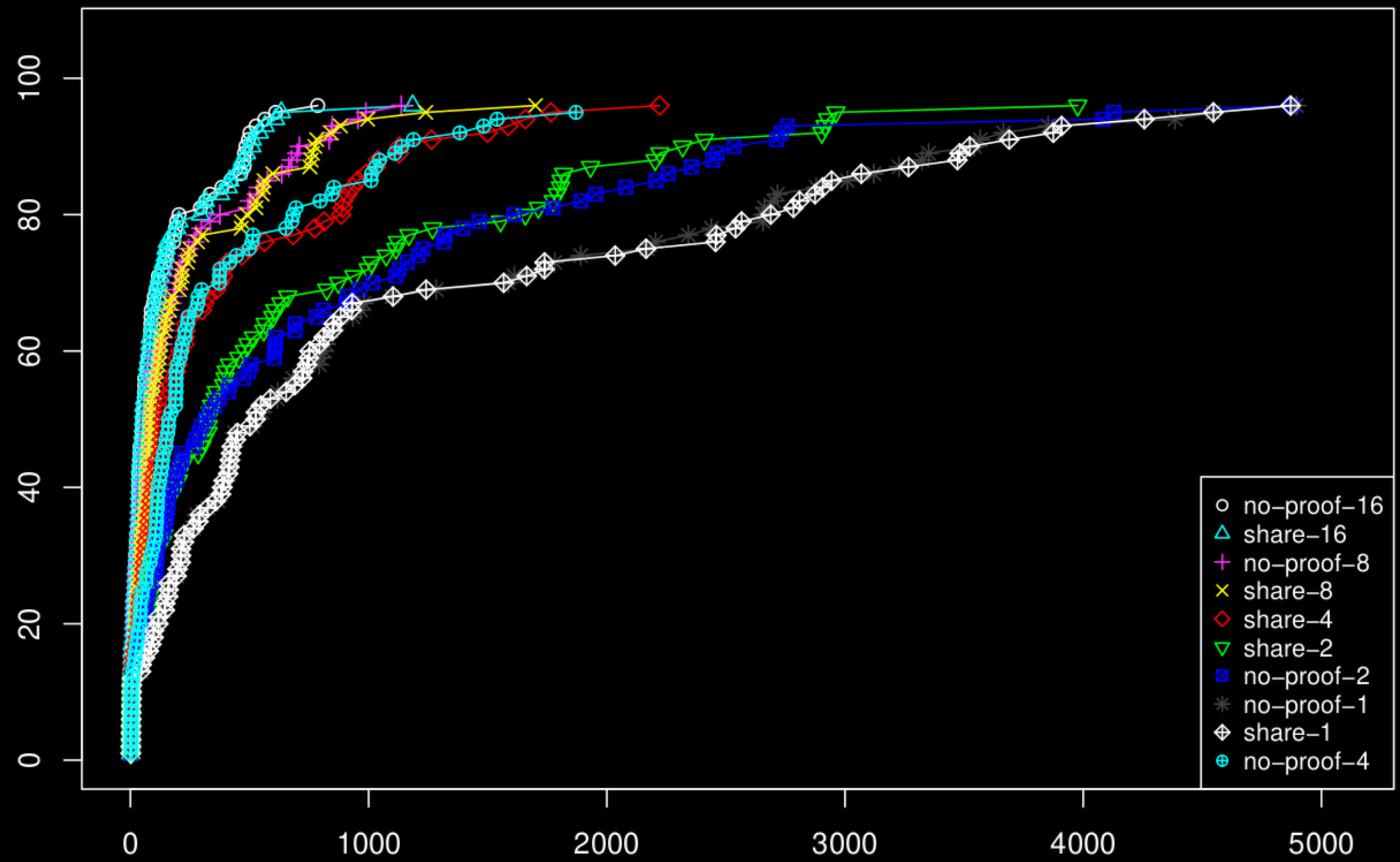


RELATIVE MEMORY INCREASE AFTER / BEFORE CLONING

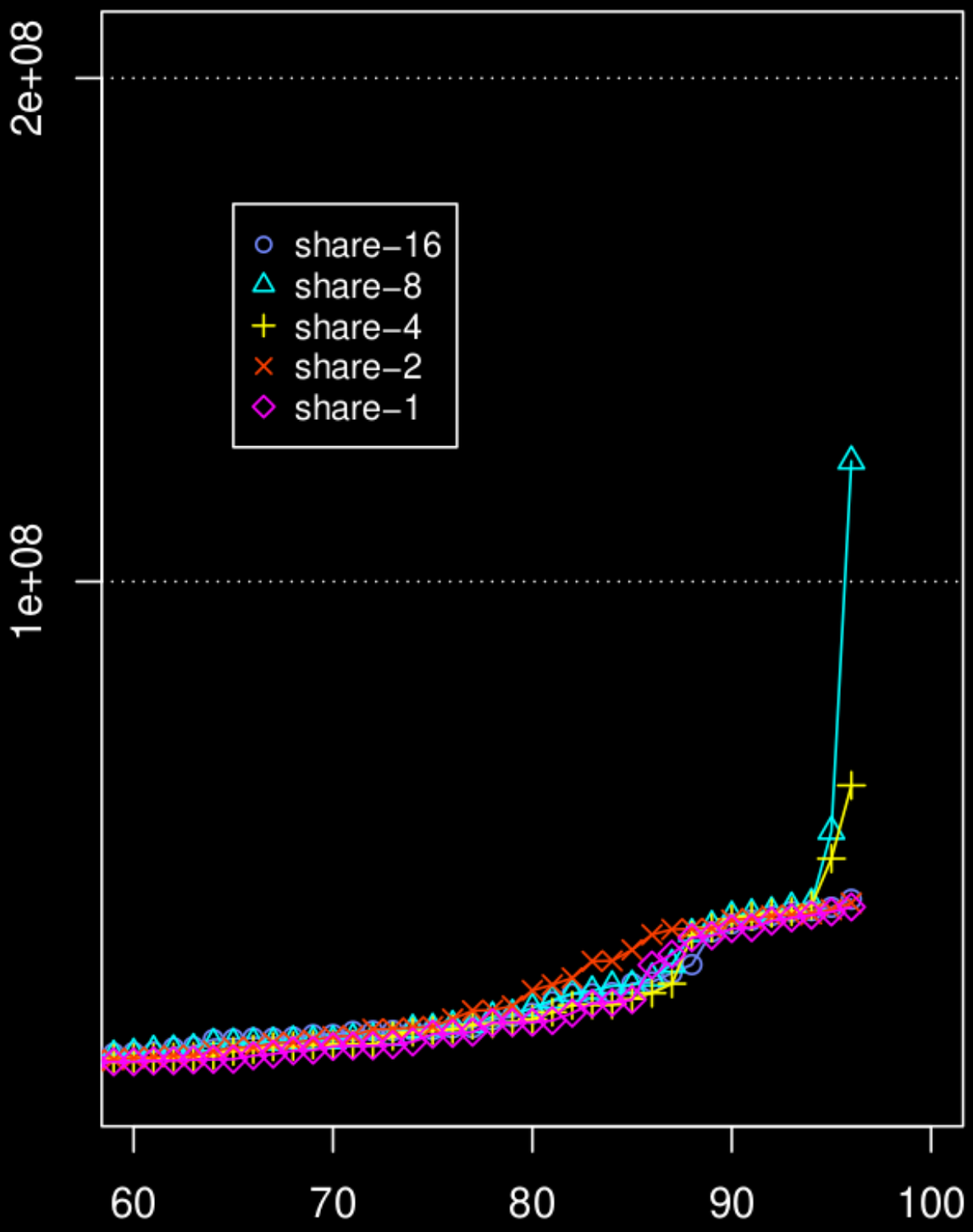


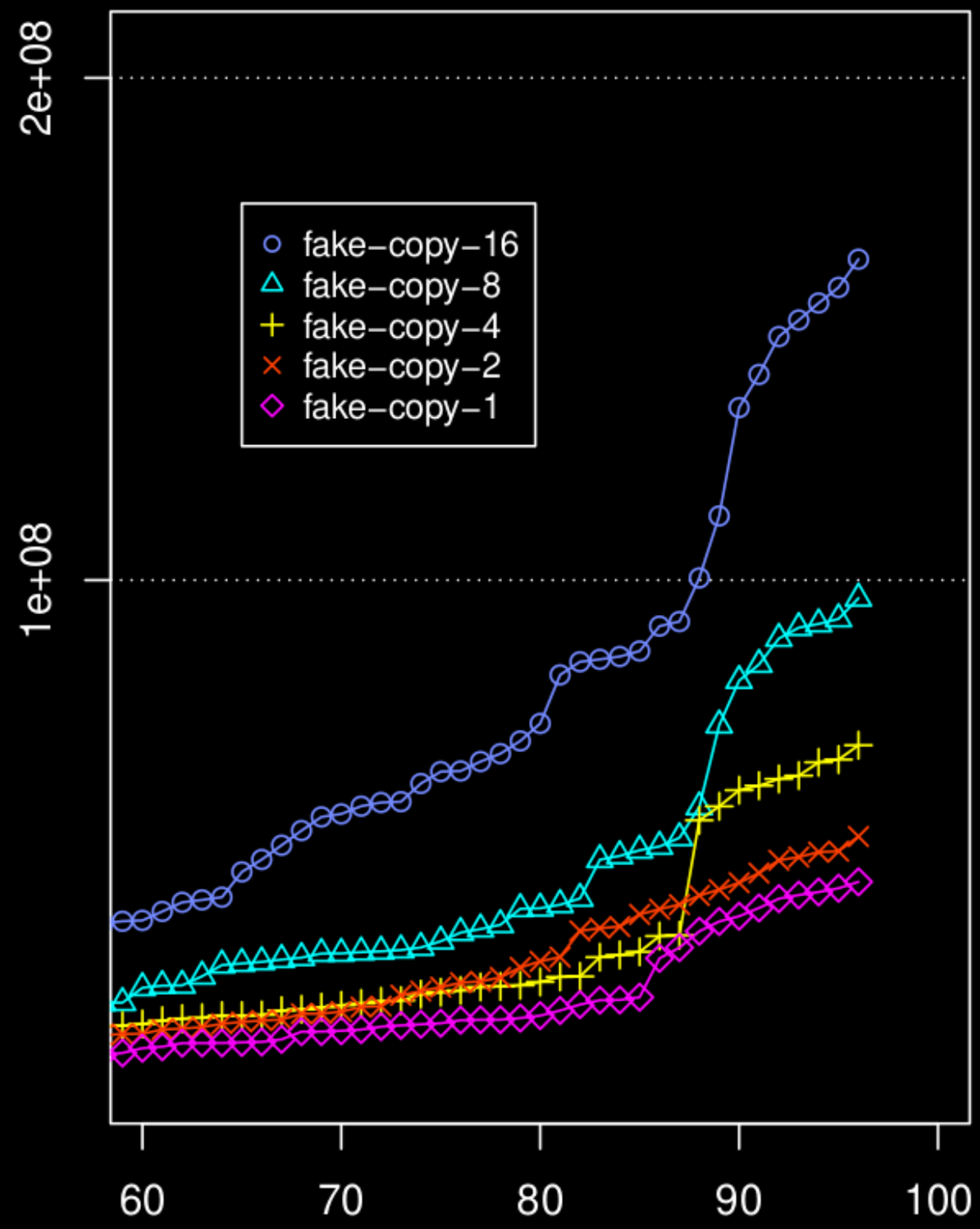
With and Without PROOF Generation

(for UNSAT formulas without inprocessing)

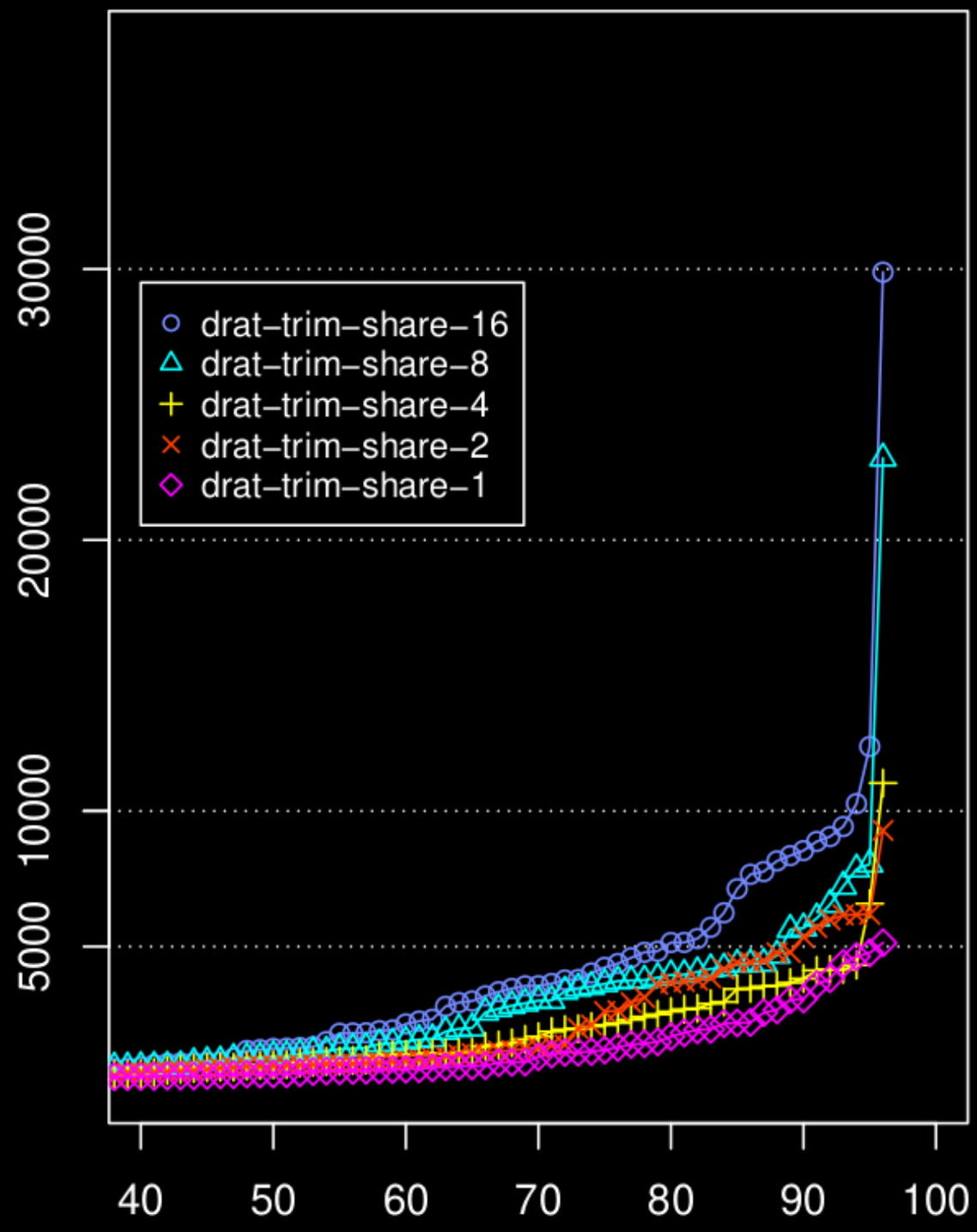


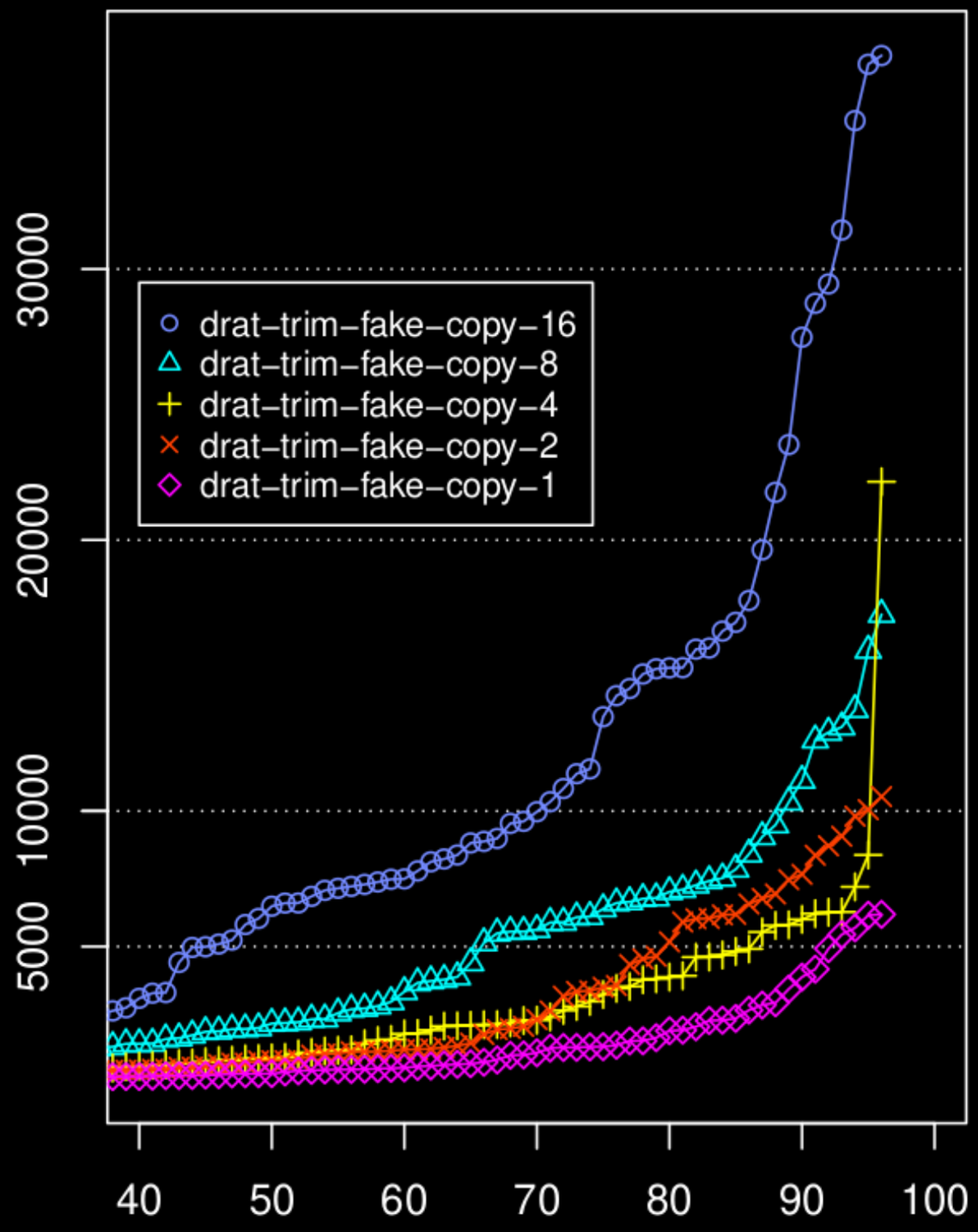
Proof Size


2e+08
1e+08
share−16
share−8
share−4
share−2
share−1
60
70
80
90
100


2e+08
1e+08
fake−copy−16
fake−copy−8
fake−copy−4
fake−copy−2
fake−copy−1
60
70
80
90
100

Proof Checking Time

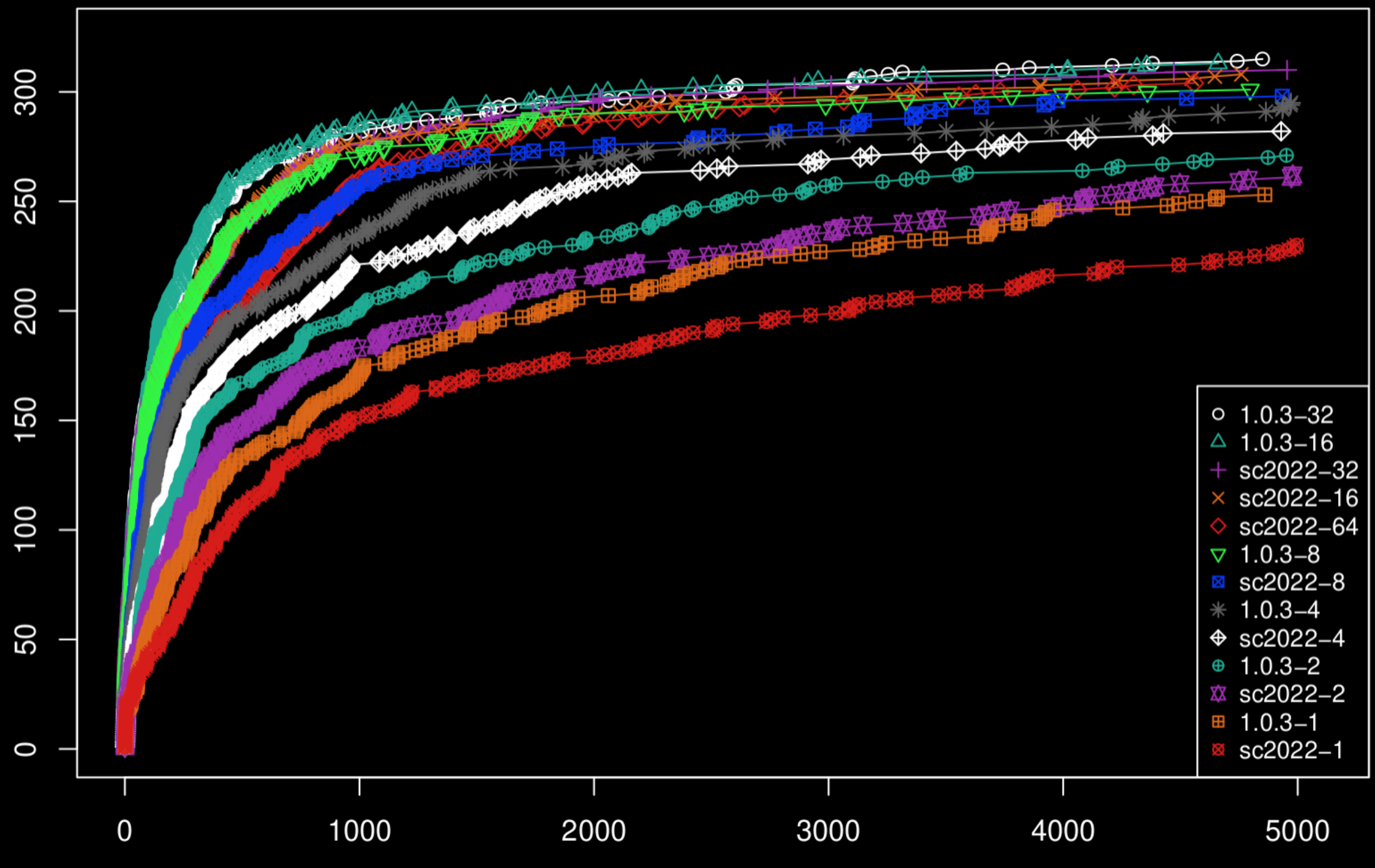




Gimsatul Version 1.0.3 vs Version SC2022

	cnt	ok	sat	uns	fld	to	mo	real	time	space	max	best	uniq
1.0.3-32	400	315	152	163	85	85	0	128894	3834670	1501822	59970	107	2
1.0.3-16	400	313	152	161	87	87	0	113829	1703738	820636	30544	83	1
sc2022-32	400	310	151	159	90	90	0	130541	3928732	1578816	63347	30	0
sc2022-16	400	308	149	159	92	92	0	135279	2069523	843444	33196	22	0
sc2022-64	400	304	150	154	96	94	2	159072	4768809	2948359	123186	16	3
1.0.3-8	400	301	144	157	99	99	0	116434	884805	446260	16887	31	0
sc2022-8	400	298	144	154	102	102	0	158110	1223199	460013	17597	9	0
1.0.3-4	400	295	141	154	105	105	0	191332	742756	263314	9978	7	1
sc2022-4	400	282	138	144	118	118	0	191043	744566	263227	10428	4	1
1.0.3-2	400	271	131	140	129	129	0	210983	415397	166175	6024	6	0
sc2022-2	400	262	130	132	138	138	0	253399	499420	167771	5896	5	0
1.0.3-1	400	253	124	129	147	147	0	256319	256291	92731	4409	4	0
sc2022-1	400	230	117	113	170	170	0	263866	263851	80947	3103	3	0

Gimsatul Version 1.0.3 vs Version SC2022



Conclusion

- [x] sharing is better than copying:
  - scales well
  - aggressive import (export)
  - substantial memory reduction
  - similar proofs (checking time/size)
- [ ] still issues:
  - parallel inprocessing / cloud
  - parallel proof checking
  - or easier to check proofs (FRAT)