



# The VLSAT-3 Benchmark Suite

**Pierre Bouvier**

**TECHNICAL  
REPORT**

**N° 0516**

December 2021

Project-Team CONVECS

ISRN INRIA/RT--0516--FR+ENG

ISSN 0249-0803





## The VLSAT-3 Benchmark Suite

Pierre Bouvier

Project-Team CONVECS

Technical Report n° 0516 — December 2021 — 20 pages

**Abstract:** This report presents VLSAT-3 (an acronym for “Very Large Boolean SATisfiability problems”), the third part of a benchmark suite to be used in scientific experiments and software competitions addressing SAT and SMT (Satisfiability Modulo Theories) solving issues. VLSAT-3 contains 1200 (600 satisfiable and 600 unsatisfiable) quantifier-free first-order logic formulas of increasing complexity, proposed in SMT-LIB format under a permissive Creative Commons license. More than 90% of these benchmarks have been used during the 16<sup>th</sup> International Satisfiability Modulo Theories Competition (SMT-COMP 2021).

**Key-words:** benchmark suite, data set, SMT-LIB, Nested-Unit Petri Net, NUPN, Petri Net, Satisfiability Modulo Theories, SMT formula, SMT solver

**RESEARCH CENTRE  
GRENOBLE – RHÔNE-ALPES**

Inovallée  
655 avenue de l'Europe Montbonnot  
38334 Saint Ismier Cedex

## Le jeu de tests VLSAT-3

**Résumé :** VLSAT-3 (acronyme anglais de “très grands problèmes de satisfaisabilité booléenne”) est le troisième volet d’une suite de tests destinée aux expérimentations scientifiques et aux compétitions de logiciels pour la résolution de problèmes SAT et SMT (Satisfaisabilité Modulo des Théories). VLSAT-3 contient 1200 formules logiques (600 satisfaisables et 600 insatisfaisables) du premier ordre sans quantificateur, de complexité croissante, fournies en format SMT-LIB sous une licence Creative Commons permissive. Plus de 90% de ces tests ont été utilisés lors de la 16<sup>ème</sup> Compétition Internationale de Satisfaisabilité Modulo des Théories (SMT-COMP 2021).

**Mots-clés :** ensemble de données, formule SMT, Nested-Unit Petri Net, NUPN, réseau de Petri, résolveur SMT, satisfaisabilité modulo des théories, SMT-LIB, suite de tests

## 1 Benchmark Description

We previously published two test suites, named VLSAT-1 [2] and VLSAT-2 [3], containing SAT formulas. VLSAT-3<sup>1</sup> is a collection of 1200 SMT (Satisfiability Modulo Theories) formulas (i.e., first-order logic formulas), written in six different quantifier-free logic fragments. For each of these logic fragments, 100 satisfiable and 100 unsatisfiable formulas are provided, which results in 12 families containing 100 benchmarks each.

Each formula is provided as a separate file, encoded in the SMT-LIB 2.6 format [1]. Each file is then compressed using bzip2 to save disk space and allow faster downloads. The 1200 formulas require 2.4 gigabytes of disk space and 132 megabytes when compressed using bzip2.

The VLSAT-3 benchmarks are licensed under the CC-BY Creative Commons Attribution 4.0 International License<sup>2</sup>.

## 2 Scientific Context

Interesting SMT formulas can be generated as a by-product of our recent work [4] on the decomposition of Petri nets into networks of automata, a problem that has been around since the early 70s. Concretely, we developed a tool chain that takes as input a Petri net (which must be ordinary, safe, and hopefully not too large) and produces as output a network of automata that execute concurrently and synchronize using shared transitions. Precisely, this network is expressed as a *Nested-Unit Petri Net* (NUPN) [5], i.e., an extension of a Petri net, in which places are grouped into sets (called *units*) that denote sequential components. A NUPN provides a proper structuring of its underlying Petri net, and enables formal verification tools to be more efficient in terms of memory and CPU time. Hence, the NUPN concept has been implemented in many tools and adopted by software competitions, such as the Model Checking Contest<sup>3</sup> [9, 8] and the Rigorous Examination of Reactive Systems challenge<sup>4</sup> [6, 10, 7]. Each NUPN generated by our tool chain is *flat*, meaning that its units are not recursively nested in each other, and *unit-safe*, meaning that each unit has at most one execution token at a time.

Our tool chain works by reformulating concurrency constraints on Petri nets as logical problems, which can be later solved using third-party software, such as SAT solvers, SMT solvers, and tools for graph coloring and finding maximum cliques, depending on the chosen strategy [4]. When a strategy involving SMT solving is selected, the tool chain produces formulas to be processed by SMT solvers. The tool chain is solver-agnostic and supports six standard SMT logic fragments.

## 3 Structure of Formulas

Each of our formulas was produced for a particular Petri net. A formula depends on four factors:

- the set  $P$  of the places of the Petri net;
- a *concurrency relation*  $\parallel$  defined over  $P$ , such that  $p \parallel p'$  iff both places  $p$  and  $p'$  may simultaneously have an execution token;

---

<sup>1</sup><https://cadp.inria.fr/resources/vlsat/3.html>

<sup>2</sup>License terms available from <http://creativecommons.org/licenses/by/4.0>

<sup>3</sup><https://mcc.lip6.fr>

<sup>4</sup><http://rers-challenge.org>

- a chosen number  $n$  of units; and
- a chosen SMT logic fragment, among QF\_BV, QF\_DT, QF\_IDL, QF\_UFBV, QF\_UFDT, and QF\_UFIDL.

A formula checks whether there exists a partition of  $P$  into  $n$  subsets  $P_i$  ( $1 \leq i \leq n$ ) such that, for each  $i$ , and for any two places  $p$  and  $p'$  of  $P_i$ ,  $p \neq p' \implies \neg(p \parallel p')$ . A model of this formula is thus an allocation of places into  $n$  leaf units, i.e., a valid decomposition of the Petri net. This can also be seen as an instance of the graph coloring problem, in which  $n$  colors are to be used for the graph with vertices defined by the places of  $P$  and edges defined by the concurrency relation. A formula is only satisfiable if the value of  $n$  is large enough (namely, greater than or equal to the chromatic number of the graph), so that at least one decomposition exists.

Notice that if a decomposition having  $n$  units exists, it is possible to generate  $n! - 1$  another similar decompositions, just by permuting unit numbers. Thus, instead of adding constraints in the formulas to express that each place  $p$  belongs to some unit of  $1 \dots n$ , we break the symmetry between units by constraining each place  $p$  to belongs some unit of  $1 \dots \min(p, n)$ .

More precisely, each formula is generated as follows, depending on the chosen logic fragment:

- The QF\_BV fragment corresponds to *quantifier-free bit-vector* logic. It supports fixed-size Boolean vectors, as well as logical, relational, and arithmetical operators on these vectors. Our encoding for QF\_BV creates, for each place  $p$ , a bit vector  $b_p$  of length  $n$  such that  $b_p[u]$  is true iff place  $p$  can belong to unit  $u$ . Then, to prevent concurrent places being assigned in the same unit, the following constraint is added: for each pair of places  $(p_1, p_2)$  such that  $(p_1 \parallel p_2) \wedge (p_1 < p_2)$ ,  $b_{p_1} \& b_{p_2} = 0_{(n)}$ , where “&” denotes the bitwise “and” operator, and “ $0_{(n)}$ ” denotes the zero vector of size  $n$ . Finally, for each place  $p$ , we add the following symmetry-breaking constraint to express that  $p$  belongs to at least one unit:  $\bigvee_{1 \leq u \leq \min(p, n)} (b_p[1 : u] \neq 0_{(u)})$ , where  $[i : j]$  denotes the operator that extracts, from a vector, the  $i^{\text{th}}$  bit to the  $j^{\text{th}}$ . Notice that there is no constraint requiring that each place belongs to one single unit. Thus, the model returned by the SMT solver may not be a partition of  $P$ . Yet, having a partition is necessary to produce a valid decomposition of the Petri net into a flat NUPN. To do so, any model that is not a partition is transformed into a partition using the first-fit-decreasing bin-packing algorithm described in [4].
- The QF\_UFBV fragment corresponds to *quantifier-free uninterpreted-function bit-vector* logic. The *uninterpreted function* theory enables the declaration of function symbols, which are given only by their signatures (i.e., the types of their arguments and results). Our encoding for QF\_UFBV is based on that of QF\_BV but, instead of the  $b_p$  variables, we define an uninterpreted function  $u$  from the set of bitvectors of size  $\lceil \log_2(\text{card}(P)) \rceil$  to the set of bitvectors of size  $n$ , each occurrence of  $b_p$  being replaced with  $u(\lambda(\#p))$  in the constraints, where  $\#p$  is a bijection from places numbers to the interval  $1 \dots \text{card}(P)$ , and where  $\lambda$  is an injection from  $1 \dots \text{card}(P)$  to the set of bitvectors of size  $\lceil \log_2(\text{card}(P)) \rceil$ .
- The QF\_DT fragment corresponds to *quantifier-free data-type* logic. It supports the definition of algebraic data types, such as enumerated types, records, lists, trees, etc. Our encoding for QF\_DT defines an enumerated type *Unit*, which contains one value per unit. Our encoding also creates, for each place  $p$ , one variable  $x_p$  of type *Unit*. Then, to prevent concurrent places being assigned to the same unit, the following constraint is added: for each pair of places  $(p_1, p_2)$  such that  $(p_1 \parallel p_2) \wedge (p_1 < p_2)$ ,  $x_{p_1} \neq x_{p_2}$ . Finally, the symmetry is broken by adding, for each variable  $x_p$  whose place number  $\#p$  is less than  $n$ , the constraint  $\bigvee_{1 \leq u \leq p} (x_p = u)$ .

- The QF\_UFDT fragment corresponds to *quantifier-free uninterpreted-function data-type* logic. Our encoding for QF\_UFDT is based on that of QF\_DT but, instead of the  $x_p$  variables, defines both an enumerated type *Place*, which contains one value per place, and an uninterpreted function  $u : Place \mapsto Unit$ , each occurrence of  $x_p$  being replaced with  $u(p)$  in the constraints.
- The QF\_IDL fragment corresponds to *quantifier-free integer-difference* logic. It supports integer variables and arithmetic constraints on the difference between two variables. The *integer difference logic* theory provides integers, which can also have constraints of the form  $(x - y) \text{ op } c$ , where  $x$  is an integer variable,  $y$  is either an integer variable or a constant,  $\text{op}$  is a comparison operator, and  $c$  is an integer constant. Our encoding for QF\_IDL is based on that of QF\_DT but declares the variables  $x_p$  to be of Integer type instead of *Unit*. Since integers are unbounded, each variable  $x_p$  whose place number  $\#p$  is greater or equal than  $n$  must be constrained by adding  $\bigvee_{1 \leq u \leq n} (x_p = u)$ , since  $x_p$  is not subject to a symmetry-break constraint. Notice that we could replace some disjunctive clauses with difference constraints (e.g.,  $x_3 = 1 \vee x_3 = 2 \vee x_3 = 3$  being replaced with  $0 \leq x_3 \leq 3$ ); this variant was tried, but found to be slower during our early experiments using the Z3 solver, and therefore, not used later.
- The QF\_UFIDL fragment corresponds to *quantifier-free uninterpreted-function integer-difference* logic. Our encoding for QF\_UFIDL is based on that of QF\_IDL, with the same changes as for evolving from QF\_BV to QF\_UFBV.

## 4 Selection of Benchmarks

We applied our approach to a large collection of more than 12,000 Petri nets from multiple sources, many of which are related to industrial problems, such as communication protocols, distributed systems, and hardware circuits. We thus generated a large collection of more than 51,000 SMT formulas produced by our tool chain. In this collection, we carefully selected a subset of formulas matching the requirements of the 16<sup>th</sup> International Satisfiability Modulo Theories Competition (SMT-COMP 2021)<sup>5</sup>.

For our experiments, we used six state-of-the-art solvers: Boolector 3.2.0 (compiled with its “--only-cadical” option), Bitwuzla (the version submitted to the SMT-COMP 2020), CVC4 1.8, MathSAT 5.6.5, Yices 2.6.2, and Z3 4.8.9. These include all the solvers that participated in SMT-COMP 2019 and/or in SMT-COMP 2020 and have been the fastest or the second fastest for any logic fragment supported by our toolchain (i.e., Boolector, Bitwuzla, CVC4, Yices, and Z3), as well as the solvers used for constructing VLSAT-2 [3] and that can solve both SAT and SMT formulas (i.e., MathSAT and Z3).

For each formula  $\phi$ , let  $\min(\phi)$  the smallest execution time measured among all the six selected solvers, using a Linux server equipped with a Xeon E5-2630 v3 and 128 GB RAM. We selected the 46,000+ formulas  $\phi$  such that  $\min(\phi)$  ranges between 10 seconds and 1 hour. For each of the six supported logic fragments, we created one family of satisfiable formulas and one family of unsatisfiable formulas. We partitioned each family in (at most) 60 classes, by gathering in the same class all formulas  $\phi$  having the same  $\min(\phi)$  rounded to the nearest minute. Finally, in each class, we selected a few formulas having a low complexity (measured in terms of SMT-LIB file size), and took the same number (plus or minus one) of formulas in each class, until reaching 100 formulas per family.

<sup>5</sup><https://smt-comp.github.io/2021/>

## 5 Resulting Benchmarks

The 12 families of the VLSAT-3 benchmark suite are listed in Table 1, the columns of which have the following meaning:

- Column 1: Letter identifying the corresponding family of formulas.
- Column 2: Satisfiability (SAT or UNSAT); all formulas in a given family have the same satisfiability.
- Column 3: One out of six quantifier-free first-order logic fragment; all formulas in a given family are in the same logic fragment.
- Column 4: Percentage of the formulas in this family that have been selected by the organizers of SMT-COMP 2021.
- Column 5: Percentage of the formulas in this family among all the benchmarks of the corresponding logic in the Single Query Track (SQT) of SMT-COMP 2021.
- Column 6: Percentage of the formulas in this family among all the benchmarks of the corresponding logic in the Model Validation Track (MVT) of SMT-COMP 2021.
- Column 7: Percentage of the formulas in this family among all the benchmarks of the corresponding logic in the Unsat Core Track (UCT) of SMT-COMP 2021.

family	satisfiability	logic	selection	SQT share	MVT share	UCT share
a	UNSAT	QF_BV	76 %	0.86 %		2.65 %
b		QF_DT	100 %	49.02 %		100 %
c		QF_IDL	100 %	8.79 %		100 %
d		QF_UFBV	88 %	29.33 %		30.67 %
e		QF_UFDT	100 %	49.26 %		100 %
f		QF_UFIDL	100 %	33.33 %		88.50 %
g	SAT	QF_BV	76 %	0.86 %	1.05 %	
h		QF_DT	100 %	49.02 %		
i		QF_IDL	70 %	6.15 %	10.13 %	
j		QF_UFBV	81 %	27.0 %	21.6 %	
k		QF_UFDT	100 %	49.26 %		
l		QF_UFIDL	100 %	33.33 %	48.54 %	

Table 1: List of the 12 families of VLSAT-3 formulas

In a nutshell, 90.9 % of the VLSAT-3 benchmarks have been used by the organizers of SMT-COMP 2021. Additionally, 7.8 % of the benchmarks of the Single Query Track, 2.9 % of the benchmarks of the Model Validation Track, and 14.0 % of the benchmarks of the Unsat Core Track are VLSAT-3 benchmarks.

We now provide 12 tables that describe each family in detail. The first one, Table 2, corresponds to family “a”, and the last one, Table 13, corresponds to family “l”.

Each row of a table corresponds to one benchmark. The columns of these tables contain the following data:



- Column name: Benchmark name, of the form “vlsat3\_Fnn.smt2.bz2”, where F is the family letter, and nn is a two-digit number ranging from 00 to 99.
- Column #variables (for QF\_BV, QF\_DT, and QF\_IDL): Number of variables declared in the formula.
- Column card (for QF\_BV, QF\_DT, and QF\_IDL): Cardinality of the type of all variables declared in the formula (or  $\infty$  if the type has an infinite set of values).
- Column card<sub>in</sub>: (for QF\_UFBV, QF\_UFDT, and QF\_UFIDL): Cardinality of the domain of the uninterpreted function present in the formula (or  $\infty$  if the domain is infinite).
- Column card<sub>out</sub>: (for QF\_UFBV, QF\_UFDT, and QF\_UFIDL): Cardinality of the codomain of the uninterpreted function present in the formula (or  $\infty$  if the codomain is infinite).
- Column #asserts: Number of assertions in the formula.
- Column #ops: Number of operators in the formula.

### Acknowledgements

The experiments presented in this paper were carried out using the GRID'5000<sup>6</sup> testbed, supported by a scientific interest group hosted by INRIA and including CNRS, RENATER and several universities as well as other organizations.

---

<sup>6</sup><https://www.grid5000.fr>

Table 2: Family “a”: List of unsatisfiable VLSAT-3 formulas written in QF\_BV logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
a00	56	$2^{11}$	1463	7283	a50	140	$2^{19}$	8750	43,614
a01	80	$2^{11}$	3050	15,170	a51	114	$2^{11}$	2465	12,177
a02	80	$2^{17}$	3050	15,218	a52	90	$2^{18}$	2808	13,996
a03	56	$2^{12}$	1463	7291	a53	118	$2^{18}$	6333	31,565
a04	120	$2^{15}$	6975	34,747	a54	176	$2^{11}$	12,053	59,993
a05	140	$2^{17}$	8750	43,598	a55	47	$2^{16}$	745	3751
a06	80	$2^{18}$	3050	15,226	a56	57	$2^{13}$	1306	6512
a07	122	$2^{11}$	6963	34,651	a57	106	$2^{11}$	4140	20,568
a08	244	$2^{10}$	16,134	80,254	a58	190	$2^{15}$	13,177	65,617
a09	80	$2^{12}$	3050	15,178	a59	80	$2^{13}$	3050	15,186
a10	112	$2^{11}$	6064	30,176	a60	145	$2^{13}$	4258	21,096
a11	164	$2^{11}$	13,172	65,612	a61	418	$2^{11}$	76,832	383,404
a12	60	$2^{15}$	1407	7027	a62	274	$2^{12}$	30,943	154,255
a13	80	$2^{16}$	3050	15,210	a63	321	$2^{11}$	46,286	230,868
a14	138	$2^{11}$	9282	46,214	a64	387	$2^{11}$	61,385	306,231
a15	48	$2^{23}$	1113	5645	a65	409	$2^{11}$	76,216	380,342
a16	56	$2^{16}$	1463	7323	a66	372	$2^{11}$	65,248	325,576
a17	84	$2^{12}$	1819	9015	a67	739	$2^{11}$	263,743	1,317,317
a18	87	$2^{14}$	3180	15,830	a68	57	$2^{12}$	1306	6504
a19	123	$2^{15}$	5120	25,466	a69	95	$2^{12}$	4076	20,278
a20	141	$2^{11}$	6054	30,068	a70	143	$2^{48}$	9242	46,300
a21	87	$2^{14}$	2672	13,290	a71	187	$2^{13}$	15,773	78,587
a22	101	$2^{19}$	3524	17,562	a72	64	$2^{14}$	2003	9991
a23	120	$2^{16}$	6975	34,755	a73	94	$2^{12}$	4007	19,935
a24	70	$2^{17}$	1633	8153	a74	214	$2^{13}$	16,183	80,583
a25	77	$2^{16}$	1907	9501	a75	341	$2^{12}$	21,828	108,546
a26	118	$2^{15}$	4995	24,851	a76	124	$2^{11}$	3938	19,522
a27	43	$2^{15}$	591	2981	a77	280	$2^{13}$	37,240	185,736
a28	99	$2^{11}$	4081	20,287	a78	80	$2^{11}$	1642	8130
a29	128	$2^{11}$	4185	20,749	a79	115	$2^{12}$	5325	26,483
a30	56	$2^{13}$	1463	7299	a80	279	$2^{11}$	33,589	167,467
a31	85	$2^{14}$	2077	10,319	a81	379	$2^{11}$	36,702	182,832
a32	87	$2^{19}$	2672	13,330	a82	438	$2^{11}$	89,288	445,644
a33	139	$2^{11}$	5667	28,137	a83	74	$2^{14}$	1048	5196
a34	161	$2^{11}$	6337	31,443	a84	513	$2^{11}$	119,974	598,924
a35	155	$2^{14}$	9328	46,434	a85	73	$2^{14}$	1136	5638
a36	61	$2^{15}$	1248	6230	a86	549	$2^{11}$	70,640	352,182
a37	78	$2^{13}$	2162	10,750	a87	455	$2^{11}$	95,112	474,730
a38	168	$2^{20}$	8560	42,616	a88	519	$2^{11}$	63,046	314,272
a39	44	$2^{25}$	946	4834	a89	398	$2^{11}$	74,088	369,724
a40	130	$2^{15}$	6171	30,707	a90	589	$2^{11}$	83,903	418,417
a41	224	$2^{13}$	15,832	78,808	a91	444	$2^{11}$	90,479	451,587
a42	237	$2^{14}$	18,611	92,685	a92	374	$2^{11}$	44,026	219,462
a43	28	$2^{11}$	224	1144	a93	482	$2^{11}$	107,271	535,471
a44	58	$2^{16}$	1555	7779	a94	588	$2^{11}$	144,272	720,264
a45	139	$2^{11}$	5449	27,047	a95	236	$2^{11}$	23,702	118,118
a46	233	$2^{15}$	20,436	101,826	a96	342	$2^{11}$	50,614	252,466
a47	45	$2^{14}$	679	3409	a97	440	$2^{121}$	91,418	457,170
a48	54	$2^{16}$	928	4652	a98	318	$2^{11}$	34,220	170,544
a49	62	$2^{30}$	1849	9353	a99	202	$2^{11}$	16,451	81,931

Table 3: Family “b”: List of unsatisfiable VLSAT-3 formulas written in QF\_DT logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
b00	56	16	1422	4504	b50	74	13	1844	5686
b01	164	11	13,018	39,162	b51	141	11	5923	17,877
b02	56	12	1418	4384	b52	187	11	8484	25,560
b03	120	16	6870	20,848	b53	163	14	9451	28,533
b04	118	14	4890	14,850	b54	263	11	18,641	56,031
b05	140	18	8627	26,185	b55	108	11	2849	8655
b06	155	13	9185	27,709	b56	80	19	2988	9304
b07	48	21	1085	3673	b57	173	11	4966	15,006
b08	161	11	6186	18,666	b58	259	11	16,567	49,809
b09	228	11	25,444	76,440	b59	290	12	33,862	101,716
b10	56	17	1423	4539	b60	312	14	34,861	104,763
b11	122	11	3633	11,007	b61	100	11	2539	7725
b12	120	17	6871	20,883	b62	182	12	15,492	46,606
b13	72	13	1947	5995	b63	197	12	17,102	51,436
b14	80	17	2986	9228	b64	121	12	4860	14,710
b15	249	15	23,529	70,795	b65	146	12	9804	29,542
b16	168	20	8411	25,611	b66	218	12	22,476	67,558
b17	155	14	9186	27,738	b67	137	21	7951	24,271
b18	192	11	9254	27,870	b68	272	11	18,181	54,651
b19	70	15	1577	4939	b69	266	12	33,804	101,542
b20	71	11	1859	5685	b70	140	12	7273	21,949
b21	80	13	2982	9100	b71	142	12	9392	28,306
b22	137	11	4890	14,778	b72	154	12	11,085	33,385
b23	112	12	5963	18,019	b73	196	12	18,722	56,296
b24	231	11	17,648	53,052	b74	265	11	23,519	70,665
b25	56	13	1419	4411	b75	271	12	33,783	101,479
b26	220	15	15,636	47,116	b76	112	12	5732	17,326
b27	366	11	33,082	99,354	b77	161	17	11,412	34,506
b28	95	14	2688	8244	b78	287	12	34,877	104,761
b29	101	19	3441	10,663	b79	735	83	265,952	804,660
b30	164	12	13,019	39,187	b80	735	84	265,953	804,829
b31	124	14	5446	16,518	b81	845	11	207,417	622,359
b32	140	19	8628	26,224	b82	236	11	23,476	70,536
b33	174	21	9097	27,709	b83	575	145	156,823	491,347
b34	72	14	1948	6024	b84	735	85	265,954	805,000
b35	124	11	3797	11,499	b85	575	144	156,822	491,056
b36	118	15	4891	14,881	b86	575	146	156,824	491,640
b37	250	11	16,203	48,717	b87	575	147	156,825	491,935
b38	265	11	23,486	70,566	b88	575	148	156,826	492,232
b39	318	11	33,912	101,844	b89	735	86	265,955	805,173
b40	36	14	387	1341	b90	609	11	149,949	449,955
b41	128	11	4067	12,309	b91	735	89	265,958	805,704
b42	643	17	27,546	82,908	b92	242	12	23,786	71,488
b43	244	11	15,900	47,808	b93	575	150	156,828	492,832
b44	228	12	25,445	76,465	b94	48	31	1065	4123
b45	262	12	33,713	101,269	b95	1026	46	404,855	1,216,633
b46	132	12	8072	24,346	b96	575	149	156,827	492,531
b47	182	11	8875	26,733	b97	1016	81	509,749	1,535,725
b48	313	11	26,245	78,843	b98	1019	81	512,764	1,544,770
b49	48	22	1086	3718	b99	717	205	249,677	790,849

Table 4: Family “c”: List of unsatisfiable VLSAT-3 formulas written in QF\_IDL logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
c00	56	$\infty$	1463	5687	c50	272	$\infty$	18,443	62,243
c01	80	$\infty$	3050	10,936	c51	297	$\infty$	22,683	75,613
c02	215	$\infty$	19,420	61,226	c52	80	$\infty$	3050	11,596
c03	216	$\infty$	23,436	117,176	c53	120	$\infty$	6975	24,523
c04	56	$\infty$	1463	5773	c54	136	$\infty$	8792	33,016
c05	80	$\infty$	3050	11,072	c55	247	$\infty$	15,746	53,502
c06	80	$\infty$	3050	11,206	c56	55	$\infty$	1026	4750
c07	194	$\infty$	18,871	67,709	c57	118	$\infty$	4995	18,723
c08	84	$\infty$	1819	7765	c58	137	$\infty$	8068	27,856
c09	80	$\infty$	3050	11,338	c59	198	$\infty$	11,362	39,076
c10	124	$\infty$	5557	20,397	c60	47	$\infty$	688	3388
c11	194	$\infty$	18,871	68,359	c61	168	$\infty$	8560	31,420
c12	56	$\infty$	1463	5857	c62	240	$\infty$	13,924	48,308
c13	112	$\infty$	6064	21,144	c63	306	$\infty$	24,224	80,470
c14	132	$\infty$	8193	28,091	c64	42	$\infty$	599	2951
c15	164	$\infty$	13,172	43,924	c65	45	$\infty$	680	3352
c16	56	$\infty$	1463	6019	c66	142	$\infty$	4788	18,156
c17	136	$\infty$	8792	32,796	c67	142	$\infty$	9523	32,617
c18	366	$\infty$	33,438	108,254	c68	74	$\infty$	1906	7844
c19	262	$\infty$	33,964	108,546	c69	161	$\infty$	6337	23,335
c20	56	$\infty$	1463	5939	c70	166	$\infty$	13,507	45,289
c21	87	$\infty$	3180	12,082	c71	244	$\infty$	16,134	54,588
c22	141	$\infty$	6054	21,412	c72	265	$\infty$	23,741	77,955
c23	161	$\infty$	6337	23,039	c73	75	$\infty$	1805	7691
c24	71	$\infty$	1920	7448	c74	83	$\infty$	2524	10,120
c25	137	$\infty$	8068	27,358	c75	128	$\infty$	5689	21,145
c26	166	$\infty$	13,507	44,985	c76	325	$\infty$	15,332	54,912
c27	325	$\infty$	15,332	53,662	c77	352	$\infty$	52,053	163,107
c28	84	$\infty$	1819	7903	c78	28	$\infty$	224	1272
c29	244	$\infty$	16,134	54,124	c79	54	$\infty$	918	4390
c30	366	$\infty$	33,438	108,964	c80	129	$\infty$	2615	11,273
c31	262	$\infty$	33,964	109,044	c81	148	$\infty$	5549	20,337
c32	80	$\infty$	3050	11,468	c82	152	$\infty$	10,953	37,207
c33	142	$\infty$	9523	32,361	c83	48	$\infty$	754	3620
c34	265	$\infty$	23,741	77,449	c84	250	$\infty$	16,443	55,671
c35	228	$\infty$	25,662	83,186	c85	321	$\infty$	46,286	145,186
c36	137	$\infty$	8068	27,608	c86	342	$\infty$	50,614	158,590
c37	137	$\infty$	8068	29,066	c87	622	$\infty$	192,992	591,324
c38	285	$\infty$	20,436	68,014	c88	57	$\infty$	1306	5330
c39	228	$\infty$	25,662	83,614	c89	113	$\infty$	4010	15,598
c40	226	$\infty$	14,011	47,751	c90	96	$\infty$	3125	12,523
c41	215	$\infty$	18,071	61,249	c91	176	$\infty$	9488	34,808
c42	285	$\infty$	20,436	68,560	c92	61	$\infty$	1248	5632
c43	289	$\infty$	21,221	70,465	c93	508	$\infty$	89,854	284,590
c44	139	$\infty$	5667	20,709	c94	520	$\infty$	135,460	677,296
c45	112	$\infty$	6064	21,340	c95	52	$\infty$	1321	5527
c46	231	$\infty$	17,869	59,455	c96	40	$\infty$	793	3511
c47	289	$\infty$	21,221	71,019	c97	118	$\infty$	3673	14,347
c48	60	$\infty$	1407	5987	c98	67	$\infty$	1245	5839
c49	164	$\infty$	13,172	44,224	c99	159	$\infty$	5105	19,583

Table 5: Family “d”: List of unsatisfiable VLSAT-3 formulas written in QF\_UFBV logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
d00	2 <sup>7</sup>	2 <sup>17</sup>	3050	21,142	d50	2 <sup>7</sup>	2 <sup>13</sup>	2508	17,319
d01	2 <sup>8</sup>	2 <sup>14</sup>	8576	59,647	d51	2 <sup>8</sup>	2 <sup>13</sup>	7304	50,723
d02	2 <sup>7</sup>	2 <sup>16</sup>	6975	48,495	d52	2 <sup>6</sup>	2 <sup>13</sup>	591	4032
d03	2 <sup>8</sup>	2 <sup>15</sup>	8576	59,649	d53	2 <sup>7</sup>	2 <sup>12</sup>	3741	25,891
d04	2 <sup>7</sup>	2 <sup>12</sup>	1819	12,503	d54	2 <sup>6</sup>	2 <sup>21</sup>	1113	7687
d05	2 <sup>7</sup>	2 <sup>19</sup>	3050	21,146	d55	2 <sup>6</sup>	2 <sup>24</sup>	1506	10,423
d06	2 <sup>6</sup>	2 <sup>13</sup>	1407	9693	d56	2 <sup>7</sup>	2 <sup>13</sup>	6363	44,223
d07	2 <sup>6</sup>	2 <sup>16</sup>	1463	10,103	d57	2 <sup>5</sup>	2 <sup>10</sup>	143	953
d08	2 <sup>7</sup>	2 <sup>11</sup>	5833	40,515	d58	2 <sup>7</sup>	2 <sup>10</sup>	551	3734
d09	2 <sup>8</sup>	2 <sup>11</sup>	9282	64,580	d59	2 <sup>8</sup>	2 <sup>10</sup>	4685	32,333
d10	2 <sup>6</sup>	2 <sup>13</sup>	1463	10,097	d60	2 <sup>7</sup>	2 <sup>11</sup>	2609	17,962
d11	2 <sup>8</sup>	2 <sup>11</sup>	9523	66,255	d61	2 <sup>8</sup>	2 <sup>11</sup>	18,157	126,483
d12	2 <sup>7</sup>	2 <sup>13</sup>	3050	21,134	d62	2 <sup>9</sup>	2 <sup>10</sup>	100,629	702,915
d13	2 <sup>8</sup>	2 <sup>11</sup>	13,507	94,071	d63	2 <sup>7</sup>	2 <sup>11</sup>	6064	42,132
d14	2 <sup>7</sup>	2 <sup>12</sup>	3050	21,132	d64	2 <sup>8</sup>	2 <sup>11</sup>	10,023	69,725
d15	2 <sup>8</sup>	2 <sup>11</sup>	11,362	78,960	d65	2 <sup>8</sup>	2 <sup>11</sup>	13,813	96,093
d16	2 <sup>6</sup>	2 <sup>15</sup>	1463	10,101	d66	2 <sup>6</sup>	2 <sup>14</sup>	585	4001
d17	2 <sup>7</sup>	2 <sup>13</sup>	1819	12,505	d67	2 <sup>8</sup>	2 <sup>11</sup>	17,483	121,807
d18	2 <sup>6</sup>	2 <sup>13</sup>	1306	8995	d68	2 <sup>9</sup>	2 <sup>10</sup>	59,070	412,341
d19	2 <sup>7</sup>	2 <sup>14</sup>	1487	10,240	d69	2 <sup>6</sup>	2 <sup>13</sup>	1321	9115
d20	2 <sup>7</sup>	2 <sup>16</sup>	3050	21,140	d70	2 <sup>9</sup>	2 <sup>11</sup>	58,589	408,976
d21	2 <sup>7</sup>	2 <sup>14</sup>	5557	38,553	d71	2 <sup>8</sup>	2 <sup>11</sup>	17,079	118,979
d22	2 <sup>7</sup>	2 <sup>13</sup>	3180	22,023	d72	2 <sup>8</sup>	2 <sup>11</sup>	18,431	128,419
d23	2 <sup>8</sup>	2 <sup>21</sup>	9488	65,928	d73	2 <sup>8</sup>	2 <sup>10</sup>	22,645	157,849
d24	2 <sup>8</sup>	2 <sup>11</sup>	4788	33,110	d74	2 <sup>7</sup>	2 <sup>10</sup>	2840	19,577
d25	2 <sup>8</sup>	2 <sup>10</sup>	5667	39,270	d75	2 <sup>6</sup>	2 <sup>13</sup>	686	4688
d26	2 <sup>6</sup>	2 <sup>14</sup>	688	4701	d76	2 <sup>9</sup>	2 <sup>11</sup>	52,979	369,892
d27	2 <sup>8</sup>	2 <sup>17</sup>	8750	60,862	d77	2 <sup>9</sup>	2 <sup>10</sup>	93,919	656,005
d28	2 <sup>7</sup>	2 <sup>15</sup>	3050	21,138	d78	2 <sup>6</sup>	2 <sup>12</sup>	754	5156
d29	2 <sup>8</sup>	2 <sup>12</sup>	15,832	110,174	d79	2 <sup>7</sup>	2 <sup>10</sup>	1971	13,551
d30	2 <sup>7</sup>	2 <sup>15</sup>	1892	13,041	d80	2 <sup>7</sup>	2 <sup>11</sup>	4803	33,335
d31	2 <sup>7</sup>	2 <sup>18</sup>	3050	21,144	d81	2 <sup>6</sup>	2 <sup>13</sup>	599	4091
d32	2 <sup>8</sup>	2 <sup>11</sup>	6337	43,896	d82	2 <sup>7</sup>	2 <sup>19</sup>	1778	12,299
d33	2 <sup>8</sup>	2 <sup>11</sup>	13,924	96,768	d83	2 <sup>6</sup>	2 <sup>15</sup>	946	6518
d34	2 <sup>7</sup>	2 <sup>13</sup>	4995	34,635	d84	2 <sup>7</sup>	2 <sup>11</sup>	5325	36,950
d35	2 <sup>8</sup>	2 <sup>16</sup>	8576	59,651	d85	2 <sup>8</sup>	2 <sup>10</sup>	8768	60,959
d36	2 <sup>7</sup>	2 <sup>11</sup>	3083	21,271	d86	2 <sup>10</sup>	2 <sup>10</sup>	160,338	1,119,945
d37	2 <sup>7</sup>	2 <sup>15</sup>	6975	48,493	d87	2 <sup>7</sup>	2 <sup>13</sup>	2933	20,324
d38	2 <sup>7</sup>	2 <sup>15</sup>	1811	12,477	d88	2 <sup>7</sup>	2 <sup>12</sup>	3801	26,320
d39	2 <sup>8</sup>	2 <sup>20</sup>	9488	65,926	d89	2 <sup>7</sup>	2 <sup>12</sup>	3285	22,741
d40	2 <sup>7</sup>	2 <sup>13</sup>	2672	18,467	d90	2 <sup>7</sup>	2 <sup>11</sup>	4380	30,362
d41	2 <sup>7</sup>	2 <sup>16</sup>	3125	21,617	d91	2 <sup>8</sup>	2 <sup>10</sup>	7927	54,793
d42	2 <sup>6</sup>	2 <sup>29</sup>	1140	7892	d92	2 <sup>8</sup>	2 <sup>10</sup>	7937	54,851
d43	2 <sup>7</sup>	2 <sup>11</sup>	4007	27,787	d93	2 <sup>6</sup>	2 <sup>31</sup>	1083	7497
d44	2 <sup>6</sup>	2 <sup>12</sup>	679	4640	d94	2 <sup>7</sup>	2 <sup>10</sup>	3607	24,973
d45	2 <sup>8</sup>	2 <sup>20</sup>	8068	56,103	d95	2 <sup>8</sup>	2 <sup>11</sup>	5408	37,414
d46	2 <sup>6</sup>	2 <sup>14</sup>	396	2693	d96	2 <sup>9</sup>	2 <sup>10</sup>	30,074	209,735
d47	2 <sup>6</sup>	2 <sup>13</sup>	578	3968	d97	2 <sup>8</sup>	2 <sup>11</sup>	21,311	148,543
d48	2 <sup>7</sup>	2 <sup>24</sup>	4346	30,180	d98	2 <sup>8</sup>	2 <sup>11</sup>	22,759	158,655
d49	2 <sup>8</sup>	2 <sup>14</sup>	12,108	84,137	d99	2 <sup>9</sup>	2 <sup>10</sup>	59,140	412,804

Table 6: Family “e”: List of unsatisfiable VLSAT-3 formulas written in QF\_UFDT logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
e00	56	16	1422	7438	e50	74	13	1844	9428
e01	80	15	2984	15,205	e51	141	11	5923	29,758
e02	80	16	2985	15,253	e52	169	11	7133	35,808
e03	56	12	1418	7264	e53	201	17	12,460	62,674
e04	164	11	13,018	65,233	e54	41	14	501	2750
e05	196	11	18,721	93,748	e55	87	14	2598	13,235
e06	70	14	1576	8125	e56	174	21	9097	46,073
e07	118	14	4890	24,695	e57	155	14	9186	46,175
e08	198	10	11,173	55,980	e58	61	16	1202	6338
e09	39	14	451	2500	e59	129	12	3518	17,764
e10	57	13	1261	6513	e60	113	19	3915	20,050
e11	136	28	8683	44,493	e61	129	11	4054	20,413
e12	56	17	1423	7489	e62	100	11	2539	12,838
e13	124	13	5445	27,433	e63	118	11	2611	13,198
e14	203	14	13,126	65,875	e64	124	11	2768	13,983
e15	65	14	1435	7420	e65	135	11	3118	15,733
e16	231	11	17,648	88,383	e66	140	12	7273	36,539
e17	306	11	23,928	119,783	e67	132	12	8072	40,534
e18	34	15	382	2195	e68	182	11	8875	44,518
e19	56	13	1419	7303	e69	268	13	25,434	127,378
e20	112	12	5963	29,989	e70	96	11	2523	12,758
e21	120	17	6871	34,729	e71	176	21	9332	47,248
e22	85	14	2005	10,270	e72	628	16	21,528	107,968
e23	80	17	2986	15,304	e73	236	11	23,476	117,523
e24	220	15	15,636	78,465	e74	137	11	4890	24,593
e25	259	11	16,530	82,793	e75	212	12	10,265	51,499
e26	122	14	4662	23,555	e76	145	13	4125	20,833
e27	118	15	4891	24,740	e77	166	12	10,260	51,474
e28	140	18	8627	43,558	e78	182	12	15,492	77,634
e29	244	11	15,900	79,643	e79	228	12	25,445	127,399
e30	70	17	1579	8269	e80	32	13	236	1388
e31	72	14	1948	9985	e81	162	12	10,199	51,169
e32	161	11	6186	31,073	e82	222	11	21,186	106,073
e33	196	12	18,722	93,784	e83	32	14	493	2710
e34	80	20	2989	15,475	e84	61	19	1735	9150
e35	127	11	3885	19,568	e85	65	12	1763	8989
e36	216	11	11,924	59,763	e86	140	12	8621	43,279
e37	310	11	23,771	118,998	e87	67	11	1545	7868
e38	49	15	779	4180	e88	245	13	11,808	59,248
e39	128	11	4067	20,478	e89	309	12	21,686	108,604
e40	124	14	5446	27,475	e90	220	14	23,406	117,275
e41	166	12	13,352	66,934	e91	96	24	4273	22,145
e42	48	13	718	3798	e92	173	11	7665	38,468
e43	156	11	7050	35,393	e93	170	12	11,340	56,874
e44	168	20	8411	42,585	e94	241	12	13,389	67,119
e45	140	19	8628	43,615	e95	328	12	25,257	126,459
e46	37	13	422	2318	e96	546	11	138,321	691,748
e47	39	15	451	2540	e97	621	43	149,152	748,383
e48	45	13	646	3438	e98	202	12	16,260	81,474
e49	79	19	1960	10,275	e99	48	31	1065	6658

Table 7: Family “f”: List of unsatisfiable VLSAT-3 formulas written in QF\_UFIDL logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
f00	$\infty$	$\infty$	1463	9406	f50	$\infty$	$\infty$	7816	43,871
f01	$\infty$	$\infty$	13,172	70,315	f51	$\infty$	$\infty$	1140	8560
f02	$\infty$	$\infty$	18,871	112,070	f52	$\infty$	$\infty$	1245	8468
f03	$\infty$	$\infty$	3050	18,949	f53	$\infty$	$\infty$	1642	10,313
f04	$\infty$	$\infty$	6975	39,718	f54	$\infty$	$\infty$	320	2400
f05	$\infty$	$\infty$	8750	50,200	f55	$\infty$	$\infty$	1778	11,856
f06	$\infty$	$\infty$	1463	9151	f56	$\infty$	$\infty$	21,296	135,145
f07	$\infty$	$\infty$	3050	17,770	f57	$\infty$	$\infty$	716	4905
f08	$\infty$	$\infty$	6064	33,625	f58	$\infty$	$\infty$	2465	15,380
f09	$\infty$	$\infty$	1463	9529	f59	$\infty$	$\infty$	2609	15,596
f10	$\infty$	$\infty$	6975	40,033	f60	$\infty$	$\infty$	1017	7738
f11	$\infty$	$\infty$	13,172	70,777	f61	$\infty$	$\infty$	2473	15,700
f12	$\infty$	$\infty$	8068	47,714	f62	$\infty$	$\infty$	13,541	76,345
f13	$\infty$	$\infty$	8193	44,890	f63	$\infty$	$\infty$	2255	13,557
f14	$\infty$	$\infty$	9523	51,850	f64	$\infty$	$\infty$	3491	20,145
f15	$\infty$	$\infty$	1306	8403	f65	$\infty$	$\infty$	17,209	92,203
f16	$\infty$	$\infty$	5449	31,000	f66	$\infty$	$\infty$	688	4787
f17	$\infty$	$\infty$	25,662	135,211	f67	$\infty$	$\infty$	2029	12,948
f18	$\infty$	$\infty$	1633	10,690	f68	$\infty$	$\infty$	15,140	81,175
f19	$\infty$	$\infty$	8750	50,569	f69	$\infty$	$\infty$	1506	10,733
f20	$\infty$	$\infty$	13,507	72,514	f70	$\infty$	$\infty$	3008	18,683
f21	$\infty$	$\infty$	1133	7750	f71	$\infty$	$\infty$	17,483	93,947
f22	$\infty$	$\infty$	1463	9649	f72	$\infty$	$\infty$	7828	43,847
f23	$\infty$	$\infty$	9251	56,237	f73	$\infty$	$\infty$	13,122	71,815
f24	$\infty$	$\infty$	3050	19,135	f74	$\infty$	$\infty$	18,528	101,246
f25	$\infty$	$\infty$	3460	21,283	f75	$\infty$	$\infty$	2783	17,557
f26	$\infty$	$\infty$	17,869	95,676	f76	$\infty$	$\infty$	5350	31,205
f27	$\infty$	$\infty$	5890	33,876	f77	$\infty$	$\infty$	12,010	68,885
f28	$\infty$	$\infty$	10,953	59,310	f78	$\infty$	$\infty$	1210	8552
f29	$\infty$	$\infty$	13,415	74,535	f79	$\infty$	$\infty$	1227	8563
f30	$\infty$	$\infty$	476	3740	f80	$\infty$	$\infty$	8750	52,015
f31	$\infty$	$\infty$	1487	10,063	f81	$\infty$	$\infty$	756	5183
f32	$\infty$	$\infty$	22,639	123,370	f82	$\infty$	$\infty$	45,060	234,207
f33	$\infty$	$\infty$	568	4325	f83	$\infty$	$\infty$	46,286	241,214
f34	$\infty$	$\infty$	698	5151	f84	$\infty$	$\infty$	9895	54,409
f35	$\infty$	$\infty$	4974	28,765	f85	$\infty$	$\infty$	36,971	194,630
f36	$\infty$	$\infty$	1287	9550	f86	$\infty$	$\infty$	144,336	740,733
f37	$\infty$	$\infty$	1487	9913	f87	$\infty$	$\infty$	51,099	267,146
f38	$\infty$	$\infty$	24,224	129,551	f88	$\infty$	$\infty$	1080	7680
f39	$\infty$	$\infty$	1248	8546	f89	$\infty$	$\infty$	1321	8410
f40	$\infty$	$\infty$	1633	11,185	f90	$\infty$	$\infty$	10,563	58,460
f41	$\infty$	$\infty$	16,805	93,685	f91	$\infty$	$\infty$	11,111	61,498
f42	$\infty$	$\infty$	599	4400	f92	$\infty$	$\infty$	28,173	171,962
f43	$\infty$	$\infty$	1907	12,898	f93	$\infty$	$\infty$	31,647	167,323
f44	$\infty$	$\infty$	2318	14,684	f94	$\infty$	$\infty$	31,597	167,073
f45	$\infty$	$\infty$	918	6595	f95	$\infty$	$\infty$	36,045	212,498
f46	$\infty$	$\infty$	2477	15,670	f96	$\infty$	$\infty$	40,702	213,845
f47	$\infty$	$\infty$	16,183	88,597	f97	$\infty$	$\infty$	54,201	282,712
f48	$\infty$	$\infty$	1113	7942	f98	$\infty$	$\infty$	24,050	127,399
f49	$\infty$	$\infty$	3362	19,815	f99	$\infty$	$\infty$	25,412	136,444

Table 8: Family “g”: List of satisfiable VLSAT-3 formulas written in QF\_BV logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
g00	127	$2^{46}$	7987	40,041	g50	273	$2^{52}$	29,938	149,552
g01	238	$2^{27}$	9157	45,517	g51	299	$2^{53}$	38,293	191,283
g02	160	$2^{62}$	12,490	62,618	g52	353	$2^{52}$	45,498	227,192
g03	160	$2^{63}$	12,490	62,626	g53	459	$2^{47}$	49,154	245,220
g04	224	$2^{89}$	15,832	79,416	g54	325	$2^{66}$	52,331	261,525
g05	192	$2^{93}$	18,397	92,337	g55	120	$2^{60}$	7029	35,377
g06	200	$2^{80}$	19,735	98,907	g56	157	$2^{56}$	12,212	61,186
g07	233	$2^{52}$	23,058	115,232	g57	248	$2^{98}$	19,689	98,725
g08	310	$2^{40}$	24,071	120,047	g58	320	$2^{65}$	20,421	101,977
g09	224	$2^{84}$	24,788	124,156	g59	233	$2^{32}$	23,904	119,302
g10	273	$2^{55}$	29,938	149,576	g60	273	$2^{51}$	29,938	149,544
g11	366	$2^{41}$	33,438	166,778	g61	263	$2^{32}$	30,228	150,862
g12	277	$2^{38}$	34,144	170,462	g62	313	$2^{53}$	37,418	186,880
g13	313	$2^{52}$	37,418	186,872	g63	421	$2^{43}$	51,099	254,989
g14	313	$2^{57}$	37,418	186,912	g64	459	$2^{48}$	63,423	316,573
g15	299	$2^{52}$	38,293	191,275	g65	164	$2^{100}$	13,466	67,794
g16	353	$2^{53}$	45,498	227,200	g66	216	$2^{75}$	22,500	112,660
g17	353	$2^{54}$	45,498	227,208	g67	249	$2^{86}$	23,764	119,002
g18	353	$2^{57}$	45,498	227,232	g68	224	$2^{92}$	15,832	79,440
g19	385	$2^{39}$	45,580	227,434	g69	200	$2^{101}$	19,625	98,525
g20	321	$2^{44}$	46,286	231,132	g70	441	$2^{21}$	25,411	126,333
g21	329	$2^{49}$	52,913	264,291	g71	220	$2^{56}$	23,613	118,065
g22	451	$2^{56}$	59,422	296,648	g72	374	$2^{56}$	40,702	203,202
g23	519	$2^{53}$	63,046	314,608	g73	187	$2^{79}$	17,173	86,115
g24	455	$2^{44}$	85,677	427,819	g74	228	$2^{132}$	25,662	128,902
g25	138	$2^{69}$	9282	46,678	g75	444	$2^{47}$	47,338	236,170
g26	139	$2^{61}$	9428	47,342	g76	549	$2^{56}$	70,640	352,542
g27	201	$2^{79}$	12,645	63,447	g77	192	$2^{91}$	18,397	92,321
g28	202	$2^{71}$	15,452	77,416	g78	400	$2^{95}$	74,565	372,777
g29	194	$2^{53}$	16,918	84,618	g79	159	$2^{104}$	12,665	63,831
g30	210	$2^{73}$	17,096	85,636	g80	308	$2^{107}$	34,025	170,357
g31	210	$2^{74}$	17,096	85,644	g81	326	$2^{65}$	40,287	201,295
g32	305	$2^{62}$	18,569	92,723	g82	315	$2^{90}$	43,690	218,532
g33	228	$2^{76}$	19,386	97,074	g83	423	$2^{50}$	54,201	270,551
g34	248	$2^{99}$	19,689	98,733	g84	309	$2^{59}$	17,964	89,666
g35	341	$2^{65}$	21,828	108,970	g85	313	$2^{51}$	37,418	186,864
g36	233	$2^{54}$	23,058	115,248	g86	216	$2^{72}$	22,500	112,636
g37	219	$2^{69}$	23,292	116,566	g87	560	$2^{59}$	75,758	378,134
g38	252	$2^{51}$	31,379	156,791	g88	194	$2^{52}$	16,918	84,610
g39	308	$2^{106}$	34,025	170,349	g89	357	$2^{38}$	30,391	151,537
g40	315	$2^{91}$	43,690	218,540	g90	248	$2^{97}$	19,689	98,717
g41	415	$2^{48}$	47,966	239,376	g91	308	$2^{109}$	34,025	170,373
g42	415	$2^{49}$	47,966	239,384	g92	410	$2^{83}$	83,441	417,041
g43	473	$2^{50}$	53,819	268,541	g93	300	$2^{150}$	44,559	223,387
g44	365	$2^{50}$	60,276	301,042	g94	250	$2^{51}$	30,881	154,305
g45	428	$2^{44}$	62,937	314,173	g95	312	$2^{108}$	35,160	176,032
g46	159	$2^{68}$	11,229	56,363	g96	176	$2^{49}$	13,719	68,627
g47	160	$2^{60}$	12,490	62,602	g97	304	$2^{72}$	42,220	211,060
g48	290	$2^{59}$	16,805	83,909	g98	412	$2^{83}$	84,259	421,127
g49	200	$2^{77}$	19,735	98,883	g99	300	$2^{100}$	44,850	224,442



Table 9: Family “h”: List of satisfiable VLSAT-3 formulas written in QF\_DT logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
h00	120	84	6938	27,784	h50	279	29	17,595	53,595
h01	138	119	9497	42,531	h51	282	31	25,260	76,708
h02	138	118	9496	42,292	h52	244	32	15,921	48,753
h03	138	113	9491	41,127	h53	233	32	23,702	72,096
h04	215	124	19,328	73,234	h54	172	44	14,208	44,514
h05	200	129	19,553	75,169	h55	408	407	67,406	367,458
h06	200	130	19,554	75,430	h56	248	101	19,541	68,721
h07	194	193	18,869	93,661	h57	500	204	119,303	399,319
h08	200	137	19,561	77,313	h58	258	30	21,170	64,378
h09	200	139	19,563	77,869	h59	621	114	149,223	460,549
h10	194	173	18,849	86,301	h60	233	32	23,648	71,934
h11	628	210	21,722	109,054	h61	302	210	42,839	172,405
h12	244	99	22,493	77,179	h62	258	31	21,171	64,441
h13	155	53	9225	30,429	h63	500	202	119,301	398,503
h14	214	80	16,048	54,462	h64	258	40	21,180	65,098
h15	237	101	18,474	65,520	h65	293	119	32,621	111,903
h16	175	35	12,096	37,476	h66	138	69	9212	32,326
h17	263	29	27,162	82,296	h67	316	243	49,213	206,443
h18	231	37	18,499	56,827	h68	258	39	21,179	65,017
h19	271	54	33,926	104,638	h69	258	41	21,181	65,181
h20	231	36	17,472	53,674	h70	284	26	28,792	87,024
h21	258	43	21,183	65,353	h71	500	207	119,306	400,558
h22	189	66	10,287	35,149	h72	260	53	13,333	42,753
h23	296	36	23,011	70,291	h73	500	210	119,309	401,815
h24	259	32	16,551	50,643	h74	258	33	21,173	64,573
h25	231	36	18,498	56,752	h75	283	57	32,424	100,462
h26	157	46	10,815	34,513	h76	319	222	45,664	186,052
h27	282	162	37,101	137,383	h77	408	124	57,697	188,341
h28	245	50	11,845	37,983	h78	961	62	446,926	1,344,558
h29	172	44	14,142	44,316	h79	961	72	446,936	1,345,918
h30	180	36	12,786	39,616	h80	333	37	33,496	101,818
h31	324	37	29,130	88,720	h81	500	228	119,327	409,735
h32	154	56	11,129	36,465	h82	224	50	21,418	66,702
h33	154	57	11,130	36,580	h83	500	226	119,325	408,823
h34	260	259	33,463	167,209	h84	159	68	11,137	37,965
h35	288	195	37,422	150,094	h85	500	227	119,326	409,278
h36	266	114	23,530	83,470	h86	501	500	105,952	567,354
h37	287	117	31,277	107,401	h87	500	225	119,324	408,370
h38	385	53	45,247	138,495	h88	272	31	18,201	55,531
h39	323	135	48,601	163,891	h89	467	466	55,858	384,262
h40	201	79	12,522	43,726	h90	500	218	119,317	405,255
h41	270	28	31,245	94,489	h91	500	251	119,350	420,798
h42	143	52	9150	30,100	h92	258	34	21,174	64,642
h43	265	29	23,504	71,322	h93	500	237	119,336	413,938
h44	261	50	12,661	40,431	h94	258	38	21,178	64,938
h45	323	129	48,595	162,295	h95	500	238	119,337	414,415
h46	285	26	29,014	87,690	h96	194	57	16,780	53,530
h47	292	229	41,987	178,171	h97	258	36	21,176	64,786
h48	270	29	16,946	51,648	h98	500	229	119,328	410,194
h49	621	110	149,219	459,645	h99	500	233	119,332	412,050

Table 10: Family “i”: List of satisfiable VLSAT-3 formulas written in QF\_IDL logic

name	#variables	card	#asserts	#ops	name	#variables	card	#asserts	#ops
i00	215	$\infty$	19,420	62,056	i50	704	$\infty$	245,856	1,176,046
i01	236	$\infty$	19,484	61,252	i51	802	$\infty$	309,062	1,427,170
i02	228	$\infty$	19,386	86,806	i52	800	$\infty$	314,447	1,512,047
i03	200	$\infty$	19,625	88,973	i53	252	$\infty$	31,647	139,789
i04	160	$\infty$	12,490	53,526	i54	511	$\infty$	130,476	653,056
i05	273	$\infty$	29,938	115,552	i55	834	$\infty$	218,412	683,318
i06	150	$\infty$	11,175	46,073	i56	813	$\infty$	300,190	1,493,444
i07	191	$\infty$	12,636	43,778	i57	926	$\infty$	378,528	1,795,514
i08	270	$\infty$	17,901	87,865	i58	224	$\infty$	24,788	104,740
i09	159	$\infty$	11,229	50,753	i59	499	$\infty$	124,750	623,748
i10	162	$\infty$	13,122	59,128	i60	606	$\infty$	183,500	908,440
i11	287	$\infty$	31,448	141,088	i61	444	$\infty$	47,338	181,586
i12	244	$\infty$	19,020	62,316	i62	813	$\infty$	300,190	1,520,938
i13	315	$\infty$	47,970	197,438	i63	800	$\infty$	314,447	1,482,059
i14	489	$\infty$	55,884	214,100	i64	252	$\infty$	31,597	133,105
i15	112	$\infty$	6064	27,186	i65	756	$\infty$	153,961	599,233
i16	144	$\infty$	8176	30,886	i66	736	$\infty$	267,936	1,288,878
i17	159	$\infty$	12,665	60,353	i67	752	$\infty$	282,287	1,391,359
i18	252	$\infty$	31,379	117,289	i68	859	$\infty$	298,000	996,732
i19	527	$\infty$	87,908	324,356	i69	709	$\infty$	130,363	526,635
i20	625	$\infty$	96,444	381,918	i70	622	$\infty$	192,992	941,358
i21	584	$\infty$	84,282	324,604	i71	762	$\infty$	290,182	1,451,944
i22	584	$\infty$	84,282	335,912	i72	830	$\infty$	342,480	1,672,436
i23	480	$\infty$	92,070	290,398	i73	919	$\infty$	361,735	1,362,921
i24	249	$\infty$	23,764	106,808	i74	498	$\infty$	124,202	619,214
i25	502	$\infty$	60,716	232,602	i75	500	$\infty$	125,250	626,248
i26	589	$\infty$	83,903	320,961	i76	520	$\infty$	135,460	677,298
i27	559	$\infty$	77,529	301,095	i77	830	$\infty$	342,480	1,717,166
i28	579	$\infty$	78,666	300,896	i78	511	$\infty$	130,449	647,427
i29	625	$\infty$	96,444	384,100	i79	830	$\infty$	338,025	1,634,371
i30	170	$\infty$	12,010	48,068	i80	756	$\infty$	153,961	600,551
i31	329	$\infty$	52,913	188,627	i81	802	$\infty$	321,602	1,608,804
i32	625	$\infty$	96,444	383,010	i82	818	$\infty$	324,914	1,566,002
i33	224	$\infty$	24,788	105,022	i83	250	$\infty$	30,881	115,591
i34	498	$\infty$	124,202	619,126	i84	752	$\infty$	282,437	1,413,559
i35	498	$\infty$	124,202	619,384	i85	216	$\infty$	22,500	94,064
i36	260	$\infty$	33,465	161,933	i86	834	$\infty$	347,625	1,739,261
i37	277	$\infty$	34,144	122,076	i87	798	$\infty$	168,909	678,521
i38	623	$\infty$	95,106	375,406	i88	549	$\infty$	70,640	270,326
i39	623	$\infty$	95,106	376,498	i89	575	$\infty$	165,568	827,900
i40	498	$\infty$	124,202	621,104	i90	365	$\infty$	60,366	215,146
i41	524	$\infty$	137,318	677,740	i91	1098	$\infty$	303,971	981,191
i42	405	$\infty$	76,926	378,694	i92	560	$\infty$	75,758	289,930
i43	645	$\infty$	102,997	408,127	i93	804	$\infty$	293,357	1,210,883
i44	645	$\infty$	102,997	411,499	i94	490	$\infty$	120,295	601,473
i45	219	$\infty$	23,292	94,798	i95	160	$\infty$	12,490	53,328
i46	409	$\infty$	76,114	271,068	i96	1081	$\infty$	451,115	1,364,133
i47	615	$\infty$	94,241	368,339	i97	1329	$\infty$	509,080	1,540,508
i48	164	$\infty$	13,466	63,296	i98	575	$\infty$	164,438	751,882
i49	606	$\infty$	183,500	908,240	i99	1324	$\infty$	497,123	1,501,947

Table 11: Family “j”: List of satisfiable VLSAT-3 formulas written in QF\_UFBV logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
j00	2 <sup>6</sup>	2 <sup>30</sup>	1000	6908	j50	2 <sup>10</sup>	2 <sup>77</sup>	62,937	439,427
j01	2 <sup>6</sup>	2 <sup>25</sup>	1250	8648	j51	2 <sup>10</sup>	2 <sup>78</sup>	62,937	439,429
j02	2 <sup>6</sup>	2 <sup>30</sup>	1280	8853	j52	2 <sup>9</sup>	2 <sup>14</sup>	73,016	509,698
j03	2 <sup>6</sup>	2 <sup>30</sup>	1665	11,533	j53	2 <sup>9</sup>	2 <sup>56</sup>	76,114	531,681
j04	2 <sup>6</sup>	2 <sup>30</sup>	1719	11,911	j54	2 <sup>9</sup>	2 <sup>56</sup>	76,216	532,395
j05	2 <sup>7</sup>	2 <sup>40</sup>	4770	33,168	j55	2 <sup>10</sup>	2 <sup>61</sup>	77,529	541,146
j06	2 <sup>11</sup>	2 <sup>98</sup>	6048	42,200	j56	2 <sup>10</sup>	2 <sup>63</sup>	82,118	573,207
j07	2 <sup>11</sup>	2 <sup>108</sup>	7768	54,215	j57	2 <sup>10</sup>	2 <sup>62</sup>	83,903	585,676
j08	2 <sup>8</sup>	2 <sup>51</sup>	8610	59,968	j58	2 <sup>9</sup>	2 <sup>43</sup>	85,677	598,458
j09	2 <sup>8</sup>	2 <sup>32</sup>	12,394	86,226	j59	2 <sup>10</sup>	2 <sup>62</sup>	87,124	608,163
j10	2 <sup>8</sup>	2 <sup>38</sup>	16,841	117,406	j60	2 <sup>9</sup>	2 <sup>21</sup>	93,051	650,074
j11	2 <sup>8</sup>	2 <sup>93</sup>	18,397	128,387	j61	2 <sup>9</sup>	2 <sup>57</sup>	114,402	799,432
j12	2 <sup>8</sup>	2 <sup>50</sup>	18,611	129,787	j62	2 <sup>9</sup>	2 <sup>81</sup>	114,402	799,480
j13	2 <sup>8</sup>	2 <sup>100</sup>	18,675	130,323	j63	2 <sup>9</sup>	2 <sup>82</sup>	114,402	799,482
j14	2 <sup>8</sup>	2 <sup>75</sup>	18,975	132,373	j64	2 <sup>9</sup>	2 <sup>83</sup>	114,402	799,484
j15	2 <sup>8</sup>	2 <sup>120</sup>	19,495	136,013	j65	2 <sup>9</sup>	2 <sup>84</sup>	114,402	799,486
j16	2 <sup>8</sup>	2 <sup>75</sup>	19,735	137,693	j66	2 <sup>9</sup>	2 <sup>85</sup>	114,402	799,488
j17	2 <sup>8</sup>	2 <sup>199</sup>	19,900	139,099	j67	2 <sup>7</sup>	2 <sup>33</sup>	2178	15,112
j18	2 <sup>9</sup>	2 <sup>68</sup>	21,399	149,153	j68	2 <sup>7</sup>	2 <sup>37</sup>	2738	19,016
j19	2 <sup>9</sup>	2 <sup>69</sup>	21,399	149,155	j69	2 <sup>7</sup>	2 <sup>40</sup>	2970	20,628
j20	2 <sup>9</sup>	2 <sup>70</sup>	21,399	149,157	j70	2 <sup>11</sup>	2 <sup>88</sup>	5458	38,065
j21	2 <sup>9</sup>	2 <sup>71</sup>	21,399	149,159	j71	2 <sup>8</sup>	2 <sup>41</sup>	9962	69,388
j22	2 <sup>9</sup>	2 <sup>64</sup>	21,696	151,341	j72	2 <sup>8</sup>	2 <sup>104</sup>	12,665	88,384
j23	2 <sup>12</sup>	2 <sup>81</sup>	24,788	173,004	j73	2 <sup>9</sup>	2 <sup>30</sup>	21,399	149,077
j24	2 <sup>9</sup>	2 <sup>21</sup>	25,411	176,594	j74	2 <sup>8</sup>	2 <sup>56</sup>	23,613	164,741
j25	2 <sup>9</sup>	2 <sup>75</sup>	31,017	216,349	j75	2 <sup>8</sup>	2 <sup>132</sup>	25,662	179,212
j26	2 <sup>9</sup>	2 <sup>157</sup>	33,964	237,274	j76	2 <sup>9</sup>	2 <sup>72</sup>	42,220	294,770
j27	2 <sup>9</sup>	2 <sup>51</sup>	38,293	267,254	j77	2 <sup>9</sup>	2 <sup>14</sup>	75,529	527,289
j28	2 <sup>9</sup>	2 <sup>65</sup>	40,287	281,159	j78	2 <sup>10</sup>	2 <sup>68</sup>	3548	24,715
j29	2 <sup>9</sup>	2 <sup>229</sup>	43,029	300,579	j79	2 <sup>7</sup>	2 <sup>45</sup>	3924	27,286
j30	2 <sup>9</sup>	2 <sup>231</sup>	43,029	300,583	j80	2 <sup>7</sup>	2 <sup>48</sup>	5848	40,694
j31	2 <sup>9</sup>	2 <sup>232</sup>	43,029	300,585	j81	2 <sup>8</sup>	2 <sup>100</sup>	13,466	93,968
j32	2 <sup>9</sup>	2 <sup>233</sup>	43,029	300,587	j82	2 <sup>7</sup>	2 <sup>50</sup>	2745	19,073
j33	2 <sup>9</sup>	2 <sup>234</sup>	43,029	300,589	j83	2 <sup>7</sup>	2 <sup>60</sup>	7029	48,961
j34	2 <sup>9</sup>	2 <sup>235</sup>	43,029	300,591	j84	2 <sup>7</sup>	2 <sup>64</sup>	7680	53,502
j35	2 <sup>9</sup>	2 <sup>236</sup>	43,029	300,593	j85	2 <sup>8</sup>	2 <sup>36</sup>	17,668	123,053
j36	2 <sup>9</sup>	2 <sup>237</sup>	43,029	300,595	j86	2 <sup>8</sup>	2 <sup>36</sup>	18,694	130,235
j37	2 <sup>9</sup>	2 <sup>238</sup>	43,029	300,597	j87	2 <sup>7</sup>	2 <sup>53</sup>	2956	20,565
j38	2 <sup>9</sup>	2 <sup>239</sup>	43,029	300,599	j88	2 <sup>7</sup>	2 <sup>38</sup>	3524	24,475
j39	2 <sup>9</sup>	2 <sup>240</sup>	43,029	300,601	j89	2 <sup>7</sup>	2 <sup>60</sup>	7200	50,158
j40	2 <sup>9</sup>	2 <sup>241</sup>	43,029	300,603	j90	2 <sup>9</sup>	2 <sup>150</sup>	44,559	311,311
j41	2 <sup>9</sup>	2 <sup>51</sup>	43,522	303,797	j91	2 <sup>8</sup>	2 <sup>72</sup>	20,264	141,342
j42	2 <sup>9</sup>	2 <sup>90</sup>	43,690	305,063	j92	2 <sup>9</sup>	2 <sup>41</sup>	32,521	226,344
j43	2 <sup>9</sup>	2 <sup>44</sup>	46,286	323,125	j93	2 <sup>8</sup>	2 <sup>60</sup>	11,010	76,708
j44	2 <sup>9</sup>	2 <sup>210</sup>	49,105	343,193	j94	2 <sup>12</sup>	2 <sup>80</sup>	24,788	173,002
j45	2 <sup>9</sup>	2 <sup>108</sup>	50,382	351,916	j95	2 <sup>10</sup>	2 <sup>44</sup>	62,937	439,361
j46	2 <sup>9</sup>	2 <sup>158</sup>	60,144	420,215	j96	2 <sup>7</sup>	2 <sup>53</sup>	4518	31,445
j47	2 <sup>10</sup>	2 <sup>73</sup>	62,937	439,419	j97	2 <sup>8</sup>	2 <sup>92</sup>	18,397	128,385
j48	2 <sup>10</sup>	2 <sup>74</sup>	62,937	439,421	j98	2 <sup>8</sup>	2 <sup>50</sup>	11,175	77,873
j49	2 <sup>10</sup>	2 <sup>75</sup>	62,937	439,423	j99	2 <sup>8</sup>	2 <sup>68</sup>	11,229	78,260

Table 12: Family “k”: List of satisfiable VLSAT-3 formulas written in QF\_UFDT logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
k00	138	137	9515	75,249	k50	233	32	23,648	119,664
k01	138	118	9496	67,953	k51	185	38	16,693	85,498
k02	138	115	9493	66,900	k52	258	41	21,181	108,283
k03	164	125	13,132	88,660	k53	400	399	79,897	636,890
k04	200	126	19,550	121,123	k54	621	95	149,204	759,225
k05	200	127	19,551	121,504	k55	621	112	149,221	764,529
k06	200	136	19,560	125,068	k56	201	79	12,522	71,695
k07	200	137	19,561	125,479	k57	292	225	41,798	284,140
k08	200	138	19,562	125,893	k58	159	68	11,137	62,383
k09	258	48	21,188	109,228	k59	245	47	11,165	58,974
k10	194	174	18,850	139,055	k60	187	38	17,059	87,328
k11	233	93	20,295	114,123	k61	258	30	21,170	107,095
k12	258	44	21,184	108,670	k62	233	32	23,702	119,934
k13	213	41	8455	44,653	k63	302	213	42,842	281,518
k14	155	54	9226	50,315	k64	172	44	14,208	73,790
k15	244	32	15,921	81,029	k65	292	39	21,479	109,540
k16	241	26	13,403	67,938	k66	256	16	9280	46,728
k17	266	112	23,528	136,064	k67	245	50	11,845	62,800
k18	167	35	10,978	56,605	k68	258	35	21,175	107,590
k19	154	57	11,130	60,324	k69	295	27	30,957	155,784
k20	189	66	10,287	57,738	k70	258	31	21,171	107,188
k21	258	43	21,183	108,538	k71	316	204	42,756	275,490
k22	134	53	8528	46,668	k72	316	243	49,213	333,788
k23	249	26	13,916	70,503	k73	150	50	11,074	58,945
k24	187	65	10,000	56,110	k74	279	29	17,595	89,135
k25	247	30	15,528	78,885	k75	258	32	21,172	107,284
k26	259	32	16,588	84,364	k76	258	40	21,180	108,160
k27	296	35	23,010	116,765	k77	258	42	21,182	108,409
k28	194	68	10,870	61,048	k78	365	364	65,923	527,085
k29	265	29	23,504	118,680	k79	261	50	12,661	66,880
k30	157	46	10,815	57,088	k80	308	125	33,841	192,205
k31	168	35	11,143	57,430	k81	500	216	119,315	665,803
k32	154	56	11,129	60,153	k82	500	220	119,319	668,425
k33	265	29	23,537	118,845	k83	500	229	119,328	674,500
k34	203	78	13,190	74,803	k84	500	223	119,322	670,423
k35	292	34	26,255	132,890	k85	500	226	119,325	672,448
k36	180	36	12,786	65,748	k86	500	218	119,317	667,108
k37	274	171	35,066	218,593	k87	500	222	119,321	669,754
k38	208	83	13,289	76,488	k88	315	314	49,630	394,945
k39	252	201	31,545	217,623	k89	2317	7	67,403	337,064
k40	259	32	16,438	83,614	k90	277	53	14,251	75,283
k41	270	29	16,946	85,890	k91	500	225	119,324	671,770
k42	282	31	25,260	127,633	k92	451	38	58,696	295,513
k43	260	216	33,420	236,328	k93	500	241	119,340	682,978
k44	621	88	149,197	757,293	k94	194	57	16,780	88,574
k45	621	104	149,213	761,925	k95	435	17	83,685	418,799
k46	288	197	37,424	244,644	k96	500	243	119,342	684,433
k47	265	209	34,792	238,750	k97	500	249	119,348	688,870
k48	282	192	37,131	240,279	k98	500	232	119,331	676,579
k49	172	44	14,142	73,460	k99	500	242	119,341	683,704

Table 13: Family “1”: List of satisfiable VLSAT-3 formulas written in QF\_UFIDL logic

name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops	name	card <sub>in</sub>	card <sub>out</sub>	#asserts	#ops
100	$\infty$	$\infty$	19,625	144,910	150	$\infty$	$\infty$	35,153	217,390
101	$\infty$	$\infty$	19,625	145,189	151	$\infty$	$\infty$	45,762	378,581
102	$\infty$	$\infty$	19,625	147,565	152	$\infty$	$\infty$	21,733	138,493
103	$\infty$	$\infty$	19,625	153,889	153	$\infty$	$\infty$	164,523	1,555,162
104	$\infty$	$\infty$	19,625	156,805	154	$\infty$	$\infty$	38,293	235,745
105	$\infty$	$\infty$	19,625	157,978	155	$\infty$	$\infty$	77,738	632,091
106	$\infty$	$\infty$	13,890	158,430	156	$\infty$	$\infty$	46,406	379,702
107	$\infty$	$\infty$	21,173	132,481	157	$\infty$	$\infty$	65,043	672,530
108	$\infty$	$\infty$	17,209	104,878	158	$\infty$	$\infty$	18,443	115,570
109	$\infty$	$\infty$	19,420	166,325	159	$\infty$	$\infty$	37,418	233,652
110	$\infty$	$\infty$	16,918	111,755	160	$\infty$	$\infty$	63,936	511,060
111	$\infty$	$\infty$	45,580	273,055	161	$\infty$	$\infty$	77,738	617,841
112	$\infty$	$\infty$	12,108	93,498	162	$\infty$	$\infty$	36,990	230,767
113	$\infty$	$\infty$	25,512	150,310	163	$\infty$	$\infty$	54,201	335,497
114	$\infty$	$\infty$	24,224	152,474	164	$\infty$	$\infty$	63,423	381,370
115	$\infty$	$\infty$	47,966	297,573	165	$\infty$	$\infty$	164,523	1,553,947
116	$\infty$	$\infty$	24,725	154,603	166	$\infty$	$\infty$	22,141	621,577
117	$\infty$	$\infty$	45,580	272,023	167	$\infty$	$\infty$	55,352	451,949
118	$\infty$	$\infty$	32,105	258,500	168	$\infty$	$\infty$	9117	61,366
119	$\infty$	$\infty$	47,966	299,760	169	$\infty$	$\infty$	96,444	623,123
120	$\infty$	$\infty$	32,105	250,382	170	$\infty$	$\infty$	95,106	619,200
121	$\infty$	$\infty$	32,105	261,830	171	$\infty$	$\infty$	26,548	167,332
122	$\infty$	$\infty$	5689	42,992	172	$\infty$	$\infty$	72,225	680,480
123	$\infty$	$\infty$	36,971	215,672	173	$\infty$	$\infty$	24,071	154,593
124	$\infty$	$\infty$	16,443	103,568	174	$\infty$	$\infty$	82,118	536,371
125	$\infty$	$\infty$	23,058	149,877	175	$\infty$	$\infty$	7401	56,958
126	$\infty$	$\infty$	24,071	155,403	176	$\infty$	$\infty$	12,931	81,843
127	$\infty$	$\infty$	38,293	235,010	177	$\infty$	$\infty$	56,145	466,238
128	$\infty$	$\infty$	9521	59,019	178	$\infty$	$\infty$	69,298	573,873
129	$\infty$	$\infty$	18,687	118,766	179	$\infty$	$\infty$	148,022	1,222,049
130	$\infty$	$\infty$	35,170	285,487	180	$\infty$	$\infty$	54,241	446,019
131	$\infty$	$\infty$	47,966	296,475	181	$\infty$	$\infty$	65,248	533,565
132	$\infty$	$\infty$	11,111	70,969	182	$\infty$	$\infty$	105,954	906,016
133	$\infty$	$\infty$	35,170	288,262	183	$\infty$	$\infty$	76,532	624,755
134	$\infty$	$\infty$	37,222	295,533	184	$\infty$	$\infty$	70,279	583,885
135	$\infty$	$\infty$	37,222	301,869	185	$\infty$	$\infty$	99,132	637,527
136	$\infty$	$\infty$	32,521	480,524	186	$\infty$	$\infty$	60,423	495,192
137	$\infty$	$\infty$	57,982	417,135	187	$\infty$	$\infty$	282,437	2,260,054
138	$\infty$	$\infty$	15,332	155,238	188	$\infty$	$\infty$	60,627	512,697
139	$\infty$	$\infty$	22,683	143,005	189	$\infty$	$\infty$	77,240	649,222
140	$\infty$	$\infty$	46,406	375,310	190	$\infty$	$\infty$	102,997	667,458
141	$\infty$	$\infty$	57,982	417,990	191	$\infty$	$\infty$	17,096	130,103
142	$\infty$	$\infty$	47,970	388,525	192	$\infty$	$\infty$	64,039	615,010
143	$\infty$	$\infty$	65,043	668,252	193	$\infty$	$\infty$	245,802	1,967,854
144	$\infty$	$\infty$	12,212	85,119	194	$\infty$	$\infty$	15,446	90,617
145	$\infty$	$\infty$	43,834	363,240	195	$\infty$	$\infty$	68,549	576,499
146	$\infty$	$\infty$	16,666	106,186	196	$\infty$	$\infty$	183,500	1,468,046
147	$\infty$	$\infty$	125,692	1,006,204	197	$\infty$	$\infty$	10,635	67,890
148	$\infty$	$\infty$	35,624	222,364	198	$\infty$	$\infty$	321,602	2,572,404
149	$\infty$	$\infty$	47,338	295,161	199	$\infty$	$\infty$	79,681	652,354

## References

- [1] Clark Barrett, Pascal Fontaine, and Cesare Tinelli. The SMT-LIB Standard: Version 2.6. Technical report, Department of Computer Science, The University of Iowa, Iowa City, Iowa, USA, June 2017. Available from <http://smt-lib.org>.
- [2] Pierre Bouvier and Hubert Garavel. The VLSAT-1 Benchmark Suite. Technical Report RT-0510, INRIA, Grenoble, France, November 2020. Available from <https://hal.inria.fr/hal-03007233> and <https://arxiv.org/abs/2011.11049>.
- [3] Pierre Bouvier and Hubert Garavel. The VLSAT-2 Benchmark Suite. Technical Report RT-0514, INRIA, Grenoble, France, September 2021. Available from <https://hal.inria.fr/hal-03337115> and <https://arxiv.org/abs/2110.06336>.
- [4] Pierre Bouvier, Hubert Garavel, and Hernán Ponce de León. Automatic Decomposition of Petri Nets into Automata Networks – A Synthetic Account. In Ryszard Janicki, Natalia Sidorova, and Thomas Chatain, editors, *Proceedings of the 41st International Conference on Application and Theory of Petri Nets and Concurrency (PETRI NETS'20)*, Paris, France, volume 12152 of *Lecture Notes in Computer Science*, pages 3–23. Springer, June 2020.
- [5] Hubert Garavel. Nested-Unit Petri Nets. *Journal of Logical and Algebraic Methods in Programming*, 104:60–85, April 2019.
- [6] Marc Jasper, Maximilian Fecke, Bernhard Steffen, Markus Schordan, Jeroen Meijer, Jaco van de Pol, Falk Howar, and Stephen F. Siegel. The RERS 2017 Challenge and Workshop. In Hakan Erdogmus and Klaus Havelund, editors, *Proceedings of the 24th ACM SIGSOFT International SPIN Symposium on Model Checking of Software (SPIN'17)*, Santa Barbara, CA, USA, pages 11–20. ACM, July 2017.
- [7] Marc Jasper, Malte Mues, Alnis Murtovi, Maximilian Schlüter, Falk Howar, Bernhard Steffen, Markus Schordan, Dennis Hendriks, Ramon R. H. Schiffelers, Harco Kuppens, and Frits W. Vaandrager. RERS 2019: Combining Synthesis with Real-World Models. In Dirk Beyer, Marieke Huisman, Fabrice Kordon, and Bernhard Steffen, editors, *Proceedings of the 25th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'19), Part III: TOOLympics*, Prague, Czech Republic, pages 101–115. Springer, April 2019.
- [8] Fabrice Kordon, Hubert Garavel, Lom Messan Hillah, Emmanuel Paviot-Adet, Loïg Jezequel, Francis Hulin-Hubard, Elvio Amparore, Marco Beccuti, Bernard Berthomieu, Hugues Evrard, Peter G. Jensen, Didier Le Botlan, Torsten Liebke, Jeroen Meijer, Jiří Srba, Yann Thierry-Mieg, Jaco van de Pol, and Karsten Wolf. MCC'2017 – The Seventh Model Checking Contest. *Transactions on Petri Nets and Other Models of Concurrency*, XIII:181–209, 2018.
- [9] Fabrice Kordon, Hubert Garavel, Lom Messan Hillah, Emmanuel Paviot-Adet, Loïg Jezequel, César Rodríguez, and Francis Hulin-Hubard. MCC'2015 – The Fifth Model Checking Contest. *Transactions on Petri Nets and Other Models of Concurrency*, XI:262–273, 2016.
- [10] Bernhard Steffen, Marc Jasper, Jeroen Meijer, and Jaco van de Pol. Property-Preserving Generation of Tailored Benchmark Petri Nets. In *Proceedings of the 17th International Conference on Application of Concurrency to System Design (ACSD'17)*, Zaragoza, Spain, pages 1–8. IEEE Computer Society, June 2017.



**RESEARCH CENTRE  
GRENOBLE – RHÔNE-ALPES**

Inovallée  
655 avenue de l'Europe Montbonnot  
38334 Saint Ismier Cedex

Publisher  
Inria  
Domaine de Voluceau - Rocquencourt  
BP 105 - 78153 Le Chesnay Cedex  
[inria.fr](http://inria.fr)

ISSN 0249-0803