

Armin Biere

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/1038905/publications.pdf>

Version: 2024-02-01

145
papers

6,390
citations

136950

32
h-index

85541

71
g-index

159
all docs

159
docs citations

159
times ranked

1955
citing authors

#	ARTICLE	IF	CITATIONS
1	Symbolic Model Checking without BDDs. Lecture Notes in Computer Science, 1999, , 193-207.	1.3	1,037
2	Bounded Model Checking. Advances in Computers, 2003, 58, 117-148.	1.6	490
3	Bounded Model Checking Using Satisfiability Solving. Formal Methods in System Design, 2001, 19, 7-34.	0.8	473
4	Effective Preprocessing in SAT Through Variable and Clause Elimination. Lecture Notes in Computer Science, 2005, , 61-75.	1.3	315
5	Boolector: An Efficient SMT Solver for Bit-Vectors and Arrays. Lecture Notes in Computer Science, 2009, , 174-177.	1.3	206
6	A survey of recent advances in SAT-based formal verification. International Journal on Software Tools for Technology Transfer, 2005, 7, 156-173.	1.9	192
7	PicoSAT Essentials. Journal of Satisfiability, Boolean Modeling and Computation, 2008, 4, 75-97.	1.2	178
8	Resolve and Expand. Lecture Notes in Computer Science, 2005, , 59-70.	1.3	143
9	High-level data races. Software Testing Verification and Reliability, 2003, 13, 207-227.	2.0	128
10	Linear Encodings of Bounded LTL Model Checking. Logical Methods in Computer Science, 2006, 2, .	0.4	116
11	Liveness Checking as Safety Checking. Electronic Notes in Theoretical Computer Science, 2002, 66, 160-177.	0.9	111
12	Inprocessing Rules. Lecture Notes in Computer Science, 2012, , 355-370.	1.3	108
13	Boolector 2.0. Journal of Satisfiability, Boolean Modeling and Computation, 2015, 9, 53-58.	1.2	79
14	Blocked Clause Elimination for QBF. Lecture Notes in Computer Science, 2011, , 101-115.	1.3	73
15	Verifying Safety Properties of a PowerPCâ™ Microprocessor Using Symbolic Model Checking without BDDs. Lecture Notes in Computer Science, 1999, , 60-71.	1.3	68
16	Blocked Clause Elimination. Lecture Notes in Computer Science, 2010, , 129-144.	1.3	68
17	Cube and Conquer: Guiding CDCL SAT Solvers by Lookaheads. Lecture Notes in Computer Science, 2012, , 50-65.	1.3	65
18	Fuzzing and delta-debugging SMT solvers. , 2009, , .		62

#	ARTICLE	IF	CITATIONS
19	DepQBF: A Dependency-Aware QBF Solver. Journal of Satisfiability, Boolean Modeling and Computation, 2010, 7, 71-76.	1.2	57
20	Verifying Large Multipliers by Combining SAT and Computer Algebra. , 2019, , .		57
21	Adaptive Restart Strategies for Conflict Driven SAT Solvers. , 2008, , 28-33.		50
22	Efficient reduction of finite state model checking to reachability analysis. International Journal on Software Tools for Technology Transfer, 2004, 5, 185-204.	1.9	49
23	Simple Bounded LTL Model Checking. Lecture Notes in Computer Science, 2004, , 186-200.	1.3	49
24	Column-wise verification of multipliers using computer algebra. , 2017, , .		47
25	Automated Testing and Debugging of SAT and QBF Solvers. Lecture Notes in Computer Science, 2010, , 44-57.	1.3	45
26	Clause Elimination for SAT and QSAT. Journal of Artificial Intelligence Research, 0, 53, 127-168.	7.0	42
27	Extended Resolution Proofs for Conjoining BDDs. Lecture Notes in Computer Science, 2006, , 600-611.	1.3	41
28	Resolution-Based Certificate Extraction for QBF. Lecture Notes in Computer Science, 2012, , 430-435.	1.3	41
29	Automated Reencoding of Boolean Formulas. Lecture Notes in Computer Science, 2013, , 102-117.	1.3	41
30	Minimizing Learned Clauses. Lecture Notes in Computer Science, 2009, , 237-243.	1.3	40
31	Compressing BMC Encodings with QBF. Electronic Notes in Theoretical Computer Science, 2007, 174, 45-56.	0.9	39
32	SAT Race 2015. Artificial Intelligence, 2016, 241, 45-65.	5.8	38
33	Applying static analysis to large-scale, multi-threaded Java programs. , 0, , .		36
34	Simulating Circuit-Level Simplifications on CNF. Journal of Automated Reasoning, 2012, 49, 583-619.	1.4	36
35	Btor2 , BtorMC and Boolector 3.0. Lecture Notes in Computer Science, 2018, , 587-595.	1.3	36
36	Combined Static and Dynamic Analysis. Electronic Notes in Theoretical Computer Science, 2005, 131, 3-14.	0.9	35

#	ARTICLE	IF	CITATIONS
37	A First Step Towards a Unified Proof Checker for QBF. , 2007, , 201-214.		35
38	Integrating Dependency Schemes in Search-Based QBF Solvers. Lecture Notes in Computer Science, 2010, , 158-171.	1.3	35
39	Clause Elimination Procedures for CNF Formulas. Lecture Notes in Computer Science, 2010, , 357-371.	1.3	34
40	A satisfiability procedure for quantified Boolean formulae. Discrete Applied Mathematics, 2003, 130, 291-328.	0.9	33
41	Optimization of Combinatorial Testing by Incremental SAT Solving. , 2015, , .		33
42	Evaluating CDCL Variable Scoring Schemes. Lecture Notes in Computer Science, 2015, , 405-422.	1.3	32
43	Shortest Counterexamples for Symbolic Model Checking of LTL with Past. Lecture Notes in Computer Science, 2005, , 493-509.	1.3	32
44	Efficient CNF Simplification Based on Binary Implication Graphs. Lecture Notes in Computer Science, 2011, , 201-215.	1.3	32
45	A Unified Proof System for QBF Preprocessing. Lecture Notes in Computer Science, 2014, , 91-106.	1.3	31
46	Factoring Out Assumptions to Speed Up MUS Extraction. Lecture Notes in Computer Science, 2013, , 276-292.	1.3	31
47	Simple Is Better: Efficient Bounded Model Checking for Past LTL. Lecture Notes in Computer Science, 2005, , 380-395.	1.3	30
48	Greedy combinatorial test case generation using unsatisfiable cores. , 2016, , .		30
49	Nenofex: Expanding NNF for QBF Solving. , 2008, , 196-210.		30
50	Managing SAT inconsistencies with HUMUS. , 2012, , .		30
51	Liveness Checking as Safety Checking for Infinite State Spaces. Electronic Notes in Theoretical Computer Science, 2006, 149, 79-96.	0.9	28
52	Efficient extraction of Skolem functions from QRAT proofs. , 2014, , .		27
53	Extended Resolution Proofs for Symbolic SAT Solving with Quantification. Lecture Notes in Computer Science, 2006, , 54-60.	1.3	25
54	Using Block-Local Atomicity to Detect Stale-Value Concurrency Errors. Lecture Notes in Computer Science, 2004, , 150-164.	1.3	24

#	ARTICLE	IF	CITATIONS
55	A Survey on Applications of Quantified Boolean Formulas. , 2019, , .		23
56	Short Proofs Without New Variables. Lecture Notes in Computer Science, 2017, , 130-147.	1.3	22
57	Detecting Cardinality Constraints in CNF. Lecture Notes in Computer Science, 2014, , 285-301.	1.3	21
58	JNuke: Efficient Dynamic Analysis for Java. Lecture Notes in Computer Science, 2004, , 462-465.	1.3	21
59	Enhancing Search-Based QBF Solving by Dynamic Blocked Clause Elimination. Lecture Notes in Computer Science, 2015, , 418-433.	1.3	21
60	SAT and ATPG. IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, 2002, , .	0.0	20
61	Improving and extending the algebraic approach for verifying gate-level multipliers. , 2018, , .		20
62	Complexity of Fixed-Size Bit-Vector Logics. Theory of Computing Systems, 2016, 59, 323-376.	1.1	19
63	Hardware model checking competition 2017. , 2017, , .		19
64	Counterexample-Guided Model Synthesis. Lecture Notes in Computer Science, 2017, , 264-280.	1.3	19
65	Bridging the Gap between Dual Propagation and CNF-based QBF Solving. , 2013, , .		18
66	SAT-Based Model Checking. , 2018, , 277-303.		18
67	BTOR. , 2008, , .		17
68	Efficient \mathcal{Q} -calculus model checking. Lecture Notes in Computer Science, 1997, , 468-471.	1.3	17
69	Modbat: A Model-Based API Tester for Event-Driven Systems. Lecture Notes in Computer Science, 2013, , 112-128.	1.3	17
70	Title is missing!. Formal Methods in System Design, 2002, 20, 159-186.	0.8	15
71	Multiple State and Single State Tableaux for Combining Local and Global Model Checking. Lecture Notes in Computer Science, 1999, , 163-179.	1.3	15
72	Solution Validation and Extraction for QBF Preprocessing. Journal of Automated Reasoning, 2017, 58, 97-125.	1.4	14

#	ARTICLE	IF	CITATIONS
73	Strong Extension-Free Proof Systems. <i>Journal of Automated Reasoning</i> , 2020, 64, 533-554.	1.4	14
74	Model-Based Testing for Verification Back-Ends. <i>Lecture Notes in Computer Science</i> , 2013, , 39-55.	1.3	14
75	Propagation based local search for bit-precise reasoning. <i>Formal Methods in System Design</i> , 2017, 51, 608-636.	0.8	13
76	What a Difference a Variable Makes. <i>Lecture Notes in Computer Science</i> , 2018, , 75-92.	1.3	13
77	Guided Merging of Sequence Diagrams. <i>Lecture Notes in Computer Science</i> , 2013, , 164-183.	1.3	13
78	Effective Bit-Width and Under-Approximation. <i>Lecture Notes in Computer Science</i> , 2009, , 304-311.	1.3	12
79	Hardware Model Checking Competition 2014: An Analysis and Comparison of Model Checkers and Benchmarks. <i>Journal of Satisfiability, Boolean Modeling and Computation</i> , 2016, 9, 135-172.	1.2	12
80	Super-Blocked Clauses. <i>Lecture Notes in Computer Science</i> , 2016, , 45-61.	1.3	12
81	AMulet 2.0 for Verifying Multiplier Circuits. <i>Lecture Notes in Computer Science</i> , 2021, , 357-364.	1.3	12
82	Incremental Inprocessing in SAT Solving. <i>Lecture Notes in Computer Science</i> , 2019, , 136-154.	1.3	11
83	A Compact Representation for Syntactic Dependencies in QBFs. <i>Lecture Notes in Computer Science</i> , 2009, , 398-411.	1.3	11
84	Reconstructing Solutions after Blocked Clause Elimination. <i>Lecture Notes in Computer Science</i> , 2010, , 340-345.	1.3	11
85	Lemmas on demand for the extensional theory of arrays. , 2008, , .		10
86	Precise and Complete Propagation Based Local Search for Satisfiability Modulo Theories. <i>Lecture Notes in Computer Science</i> , 2016, , 199-217.	1.3	10
87	Encoding Redundancy for Satisfaction-Driven Clause Learning. <i>Lecture Notes in Computer Science</i> , 2019, , 41-58.	1.3	10
88	Failed Literal Detection for QBF. <i>Lecture Notes in Computer Science</i> , 2011, , 259-272.	1.3	10
89	Incremental column-wise verification of arithmetic circuits using computer algebra. <i>Formal Methods in System Design</i> , 2020, 56, 22-54.	0.8	9
90	Clausal Proofs of Mutilated Chessboards. <i>Lecture Notes in Computer Science</i> , 2019, , 204-210.	1.3	9

#	ARTICLE	IF	CITATIONS
91	Verifying the IEEE 1394 FireWire Tree Identify Protocol with SMV. Formal Aspects of Computing, 2003, 14, 267-280.	1.8	8
92	SDL Versus C Equivalence Checking. Lecture Notes in Computer Science, 2005, , 323-338.	1.3	8
93	Consistency Checking of All Different Constraints over Bit-Vectors within a SAT Solver. , 2008, , .		8
94	SAT Solving with GPU Accelerated Inprocessing. Lecture Notes in Computer Science, 2021, , 133-151.	1.3	8
95	Enforcer “ Efficient Failure Injection. Lecture Notes in Computer Science, 2006, , 412-427.	1.3	8
96	Cube-and-Conquer for Satisfiability. , 2018, , 31-59.		8
97	Concurrent Cube-and-Conquer. Lecture Notes in Computer Science, 2012, , 475-476.	1.3	8
98	More on the Complexity of Quantifier-Free Fixed-Size Bit-Vector Logics with Binary Encoding. Lecture Notes in Computer Science, 2013, , 378-390.	1.3	8
99	Backing Backtracking. Lecture Notes in Computer Science, 2019, , 250-266.	1.3	7
100	Chapter 9. Preprocessing in SAT Solving. Frontiers in Artificial Intelligence and Applications, 2021, , .	0.3	7
101	Nullstellensatz-Proofs for Multiplier Verification. Lecture Notes in Computer Science, 2020, , 368-389.	1.3	7
102	Everything You Always Wanted to Know about Blocked Sets (But Were Afraid to Ask). Lecture Notes in Computer Science, 2014, , 317-332.	1.3	7
103	Evaluating CDCL Restart Schemes. , 0, , .		7
104	JVM Independent Replay in Java. Electronic Notes in Theoretical Computer Science, 2005, 113, 85-104.	0.9	6
105	Turbo-charging Lemmas on demand with don't care reasoning. , 2014, , .		6
106	Improving Implementation of SLS Solvers for SAT and New Heuristics for k-SAT with Long Clauses. Lecture Notes in Computer Science, 2014, , 302-316.	1.3	6
107	Implicit Hitting Set Algorithms for Maximum Satisfiability Modulo Theories. Lecture Notes in Computer Science, 2018, , 134-151.	1.3	6
108	Revisiting Hyper Binary Resolution. Lecture Notes in Computer Science, 2013, , 77-93.	1.3	6

#	ARTICLE	IF	CITATIONS
109	Blocked Clause Decomposition. Lecture Notes in Computer Science, 2013, , 423-438.	1.3	6
110	A comparison of strategies for tolerating inconsistencies during decision-making. , 2012, , .		6
111	Tutorial on Model Checking: Modelling and Verification in Computer Science. Lecture Notes in Computer Science, 2008, , 16-21.	1.3	6
112	Subroutine Inlining and Bytecode Abstraction to Simplify Static and Dynamic Analysis. Electronic Notes in Theoretical Computer Science, 2005, 141, 109-128.	0.9	5
113	Distributed Cube and Conquer with Paracooba. Lecture Notes in Computer Science, 2020, , 114-122.	1.3	5
114	Adding Dual Variables to Algebraic Reasoning for Gate-Level Multiplier Verification. , 2022, , .		5
115	Advanced unit testing. , 2006, , .		4
116	Efficiently Representing Existential Dependency Sets for Expansion-based QBF Solvers. Electronic Notes in Theoretical Computer Science, 2009, 251, 83-95.	0.9	4
117	Dualizing Projected Model Counting. , 2018, , .		4
118	On the Complexity of Symbolic Verification and Decision Problems in Bit-Vector Logic. Lecture Notes in Computer Science, 2014, , 481-492.	1.3	4
119	Simulating Strong Practical Proof Systems with Extended Resolution. Journal of Automated Reasoning, 2020, 64, 1247-1267.	1.4	3
120	Certifying Hardware Model Checking Results. Lecture Notes in Computer Science, 2019, , 498-502.	1.3	3
121	Experimenting with SAT Solvers in Vampire. Lecture Notes in Computer Science, 2014, , 431-442.	1.3	3
122	Efficient Model Checking of Applications with Input/Output. , 2007, , 515-522.		3
123	Better lemmas with lambda extraction. , 2015, , .		2
124	A Duality-Aware Calculus for Quantified Boolean Formulas. , 2016, , .		2
125	Progress in Certifying Hardware Model Checking Results. Lecture Notes in Computer Science, 2021, , 363-386.	1.3	2
126	Non-clausal Redundancy Properties. Lecture Notes in Computer Science, 2021, , 252-272.	1.3	2

#	ARTICLE	IF	CITATIONS
127	bv2epr: A Tool for Polynomially Translating Quantifier-Free Bit-Vector Formulas into EPR. Lecture Notes in Computer Science, 2013, , 443-449.	1.3	2
128	Exhaustive Testing of Exception Handlers with Enforcer. Lecture Notes in Computer Science, 2007, , 26-46.	1.3	2
129	Skolem Function Continuation for Quantified Boolean Formulas. Lecture Notes in Computer Science, 2017, , 129-138.	1.3	2
130	Truth Assignments as Conditional Autarkies. Lecture Notes in Computer Science, 2019, , 48-64.	1.3	2
131	Practical algebraic calculus and Nullstellensatz with the checkers Pacheck and Pastèque and Nuss-Checker. Formal Methods in System Design, 0, , 1.	0.8	2
132	Challenges in Verifying Arithmetic Circuits Using Computer Algebra. , 2017, , .		1
133	Covered Clauses Are Not Propagation Redundant. Lecture Notes in Computer Science, 2020, , 32-47.	1.3	1
134	SAT, SMT and Applications. Lecture Notes in Computer Science, 2009, , 1-1.	1.3	1
135	SmacC: A Retargetable Symbolic Execution Engine. Lecture Notes in Computer Science, 2013, , 482-486.	1.3	1
136	Four Flavors of Entailment. Lecture Notes in Computer Science, 2020, , 62-71.	1.3	1
137	Combining Local and Global Model Checking. Electronic Notes in Theoretical Computer Science, 2001, 23, 34-45.	0.9	0
138	Formal Methods Group ETH Zürich. Electronic Notes in Theoretical Computer Science, 2003, 80, 289-293.	0.9	0
139	Introductory paper. International Journal on Software Tools for Technology Transfer, 2005, 7, 87-88.	1.9	0
140	Preface to the Special Issue on Automated Reasoning Systems. Journal of Automated Reasoning, 2020, 64, 361-362.	1.4	0
141	Efficient All-UIP Learned Clause Minimization. Lecture Notes in Computer Science, 2021, , 171-187.	1.3	0
142	XOR Local Search for Boolean Brent Equations. Lecture Notes in Computer Science, 2021, , 417-435.	1.3	0
143	Améliorer SAT dans le cadre incrémental. Revue D'Intelligence Artificielle, 2014, 28, 593-614.	0.6	0
144	Duplex Encoding of Staircase At-Most-One Constraints for the Antibandwidth Problem. Lecture Notes in Computer Science, 2020, , 186-204.	1.3	0

#	ARTICLE	IF	CITATIONS
145	Combining Conflict-Driven Clause Learning and Chronological Backtracking for Propositional Model Counting. , 0, , .		0