Alessandro Cimatti

List of Publications by Year in Descending Order

Source: https://exaly.com/author-pdf/3025251/alessandro-cimatti-publications-by-year.pdf
Version: 2024-04-20

This document has been generated based on the publications and citations recorded by exaly.com. For the latest version of this publication list, visit the link given above.
The third column is the impact factor (IF) of the journal, and the fourth column is the number of citations of the article.

176 papers	5,112 citations	32 h-index	67 g-index
181 ext. papers	5,683 ext. citations	1.2 avg, IF	5.53 L-index

#	Paper	IF	Citations
176	Searching for Ribbon-Shaped Paths in Fair Transition Systems. *Lecture Notes in Computer Science*, **2022**, 543-560	0.9	1
175	A first-order logic characterisation of safety and co-safety languages. *Lecture Notes in Computer Science*, **2022**, 244-263	0.9	
174	NORMA: a tool for the analysis of Relay-based Railway Interlocking Systems. *Lecture Notes in Computer Science*, **2022**, 125-142	0.9	2
173	Efficient Analysis of Cyclic Redundancy Architectures via Boolean Fault Propagation. *Lecture Notes in Computer Science*, **2022**, 273-291	0.9	1
172	Diagnosability of Fair Transition Systems. *Artificial Intelligence*, **2022**, 103725	3.6	
171	Formal Design and Validation of an Automatic Train Operation Control System. *Lecture Notes in Computer Science*, **2022**, 169-178	0.9	1
170	Automatic Discovery of Fair Paths in Infinite-State Transition Systems. *Lecture Notes in Computer Science*, **2021**, 32-47	0.9	0
169	Assumption-Based Runtime Verification of Infinite-State Systems. *Lecture Notes in Computer Science*, **2021**, 207-227	0.9	3
168	Model-based Safety Assessment of a Triple Modular Generator with xSAP. *Formal Aspects of Computing*, **2021**, 33, 251-295	1.2	1
167	Proving the Existence of Fair Paths in Infinite-State Systems. *Lecture Notes in Computer Science*, **2021**, 104-126	0.9	2
166	Implicit Semi-Algebraic Abstraction for Polynomial Dynamical Systems. *Lecture Notes in Computer Science*, **2021**, 529-551	0.9	1
165	A Comprehensive Approach to On-board Autonomy Verification and Validation. *ACM Transactions on Intelligent Systems and Technology*, **2021**, 12, 1-29	8	
164	Efficient SMT-Based Analysis of Failure Propagation. *Lecture Notes in Computer Science*, **2021**, 209-230	0.9	2
163	Optimization Modulo Non-linear Arithmetic via Incremental Linearization. *Lecture Notes in Computer Science*, **2021**, 213-231	0.9	
162	Universal Invariant Checking of Parametric Systems with Quantifier-free SMT Reasoning. *Lecture Notes in Computer Science*, **2021**, 131-147	0.9	5
161	Synthesis of P-Stable Abstractions. *Lecture Notes in Computer Science*, **2020**, 214-230	0.9	2
160	Safe Decomposition of Startup Requirements: Verification and Synthesis. *Lecture Notes in Computer Science*, **2020**, 155-172	0.9	

159	Computation of the Transient in Max-Plus Linear Systems via SMT-Solving. *Lecture Notes in Computer Science*, **2020**, 161-177	0.9	2
158	A Model-Based Approach to the Design, Verification and Deployment of Railway Interlocking System. *Lecture Notes in Computer Science*, **2020**, 240-254	0.9	5
157	SMT-based satisfiability of first-order LTL with event freezing functions and metric operators. *Information and Computation*, **2020**, 272, 104502	0.8	4
156	Formal reliability analysis of redundancy architectures. *Formal Aspects of Computing*, **2019**, 31, 59-94	1.2	6
155	NuRV: A nuXmv Extension for Runtime Verification. *Lecture Notes in Computer Science*, **2019**, 382-392	0.9	2
154	COMPASS 3.0. *Lecture Notes in Computer Science*, **2019**, 379-385	0.9	8
153	Extending nuXmv with Timed Transition Systems and Timed Temporal Properties. *Lecture Notes in Computer Science*, **2019**, 376-386	0.9	10
152	Model-Based Run-Time Synthesis of Architectural Configurations for Adaptive MILS Systems. *Lecture Notes in Computer Science*, **2019**, 200-215	0.9	1
151	Assumption-Based Runtime Verification with Partial Observability and Resets. *Lecture Notes in Computer Science*, **2019**, 165-184	0.9	8
150	Robustness Envelopes for Temporal Plans. *Proceedings of the AAAI Conference on Artificial Intelligence*, **2019**, 33, 7538-7545	5	1
149	Formal Specification and Verification of Dynamic Parametrized Architectures. *Lecture Notes in Computer Science*, **2018**, 625-644	0.9	4
148	Tightening the contract refinements of a system architecture. *Formal Methods in System Design*, **2018**, 52, 88-116	1.4	1
147	Strong temporal planning with uncontrollable durations. *Artificial Intelligence*, **2018**, 256, 1-34	3.6	6
146	Symbolic execution with existential second-order constraints **2018**,		15
145	Analysis of Relay Interlocking Systems via SMT-based Model Checking of Switched Multi-Domain Kirchhoff Networks **2018**,		5
144	Incremental Linearization for Satisfiability and Verification Modulo Nonlinear Arithmetic and Transcendental Functions. *ACM Transactions on Computational Logic*, **2018**, 19, 1-52	0.9	16
143	Experimenting on Solving Nonlinear Integer Arithmetic with Incremental Linearization. *Lecture Notes in Computer Science*, **2018**, 383-398	0.9	8
142	Satisfiability checking and symbolic computation. *ACM Communications in Computer Algebra*, **2017**, 50, 145-147	0.2	2

141	Invariant Checking of NRA Transition Systems via Incremental Reduction to LRA with EUF. *Lecture Notes in Computer Science*, **2017**, 58-75	0.9	14
140	Timed Failure Propagation Analysis for Spacecraft Engineering: The ESA Solar Orbiter Case Study. *Lecture Notes in Computer Science*, **2017**, 255-271	0.9	2
139	Satisfiability Modulo Transcendental Functions via Incremental Linearization. *Lecture Notes in Computer Science*, **2017**, 95-113	0.9	6
138	Formal Methods for Aerospace Systems **2017**, 133-159		3
137	Model Checking at Scale: Automated Air Traffic Control Design Space Exploration. *Lecture Notes in Computer Science*, **2016**, 3-22	0.9	16
136	Infinite-State Liveness-to-Safety via Implicit Abstraction and Well-Founded Relations. *Lecture Notes in Computer Science*, **2016**, 271-291	0.9	8
135	Dynamic controllability via Timed Game Automata. *Acta Informatica*, **2016**, 53, 681-722	0.9	10
134	Tightening a Contract Refinement. *Lecture Notes in Computer Science*, **2016**, 386-402	0.9	1
133	(mathsf {SC}^mathsf{2}): Satisfiability Checking Meets Symbolic Computation. *Lecture Notes in Computer Science*, **2016**, 28-43	0.9	12
132	Model-Based Design of an Energy-System Embedded Controller Using Taste. *Lecture Notes in Computer Science*, **2016**, 741-747	0.9	2
131	The xSAP Safety Analysis Platform. *Lecture Notes in Computer Science*, **2016**, 533-539	0.9	36
130	From Electrical Switched Networks to Hybrid Automata. *Lecture Notes in Computer Science*, **2016**, 164-181	0.9	4
129	Infinite-state invariant checking with IC3 and predicate abstraction. *Formal Methods in System Design*, **2016**, 49, 190-218	1.4	19
128	A Temporal Logics Approach to Contract-Based Design **2016**,		1
127	Solving strong controllability of temporal problems with uncertainty using SMT. *Constraints*, **2015**, 20, 1-29	0.3	9
126	An SMT-based approach to weak controllability for disjunctive temporal problems with uncertainty. *Artificial Intelligence*, **2015**, 224, 1-27	3.6	5
125	Formal Verification of Infinite-State BIP Models. *Lecture Notes in Computer Science*, **2015**, 326-343	0.9	17
124	Safety assessment of AltaRica models via symbolic model checking. *Science of Computer Programming*, **2015**, 98, 464-483	1.1	19

123	Contracts-refinement proof system for component-based embedded systems. *Science of Computer Programming*, **2015**, 97, 333-348	1.1	42
122	Comparing different functional allocations in automated air traffic control design **2015**,		10
121	HyComp: An SMT-Based Model Checker for Hybrid Systems. *Lecture Notes in Computer Science*, **2015**, 52-67	0.9	36
120	HRELTL: A temporal logic for hybrid systems. *Information and Computation*, **2015**, 245, 54-71	0.8	13
119	Formal Design of Asynchronous Fault Detection and Identification Components using Temporal Epistemic Logic. *Logical Methods in Computer Science*, **2015**, 11,		10
118	Efficient Anytime Techniques for Model-Based Safety Analysis. *Lecture Notes in Computer Science*, **2015**, 603-621	0.9	18
117	Combining MILS with Contract-Based Design for Safety and Security Requirements. *Lecture Notes in Computer Science*, **2015**, 264-276	0.9	12
116	Quantifier-free encoding of invariants for hybrid systems. *Formal Methods in System Design*, **2014**, 45, 165-188	1.4	4
115	Spacecraft early design validation using formal methods. *Reliability Engineering and System Safety*, **2014**, 132, 20-35	6.3	33
114	Innovative Rover Operations Concepts - Autonomous Planner (IRONCAP) - Concluding the adventure **2014**,		1
113	Sound and Complete Algorithms for Checking the Dynamic Controllability of Temporal Networks with Uncertainty, Disjunction and Observation **2014**,		10
112	Formal Safety Assessment via Contract-Based Design. *Lecture Notes in Computer Science*, **2014**, 81-97	0.9	12
111	Requirements Refinement and Component Reuse. *Advances in Systems Analysis, Software Engineering, and High Performance Computing Book Series*, **2014**, 209-241	0.4	4
110	The nuXmv Symbolic Model Checker. *Lecture Notes in Computer Science*, **2014**, 334-342	0.9	176
109	Verifying LTL Properties of Hybrid Systems with K-Liveness. *Lecture Notes in Computer Science*, **2014**, 424-440	0.9	13
108	An Integrated Process for FDIR Design in Aerospace. *Lecture Notes in Computer Science*, **2014**, 82-95	0.9	13
107	Formal Design of Fault Detection and Identification Components Using Temporal Epistemic Logic. *Lecture Notes in Computer Science*, **2014**, 326-340	0.9	8
106	IC3 Modulo Theories via Implicit Predicate Abstraction. *Lecture Notes in Computer Science*, **2014**, 46-61	0.9	51

105	Automated Analysis of Reliability Architectures **2013**,		6
104	Time-aware relational abstractions for hybrid systems **2013**,		12
103	OCRA: A tool for checking the refinement of temporal contracts **2013**,		61
102	Parameter synthesis with IC3 **2013**,		32
101	SMT-based scenario verification for hybrid systems. *Formal Methods in System Design*, **2013**, 42, 46-66	1.4	23
100	Software Model Checking SystemC. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **2013**, 32, 774-787	2.5	27
99	The MathSAT5 SMT Solver. *Lecture Notes in Computer Science*, **2013**, 93-107	0.9	236
98	A Modular Approach to MaxSAT Modulo Theories. *Lecture Notes in Computer Science*, **2013**, 150-165	0.9	17
97	SMT-Based Software Model Checking. *Lecture Notes in Computer Science*, **2013**, 23-23	0.9	
96	Efficient Analysis of Reliability Architectures via Predicate Abstraction. *Lecture Notes in Computer Science*, **2013**, 279-294	0.9	4
95	Validation of requirements for hybrid systems. *ACM Transactions on Software Engineering and Methodology*, **2012**, 21, 1-34	3.3	25
94	A Property-Based Proof System for Contract-Based Design **2012**,		40
93	Formal Verification and Validation of ERTMS Industrial Railway Train Spacing System. *Lecture Notes in Computer Science*, **2012**, 378-393	0.9	19
92	Software Model Checking with Explicit Scheduler and Symbolic Threads. *Logical Methods in Computer Science*, **2012**, 8,		5
91	Software Model Checking via IC3. *Lecture Notes in Computer Science*, **2012**, 277-293	0.9	71
90	Solving Temporal Problems Using SMT: Strong Controllability. *Lecture Notes in Computer Science*, **2012**, 248-264	0.9	7
89	Safety, Dependability and Performance Analysis of Extended AADL Models. *Computer Journal*, **2011** , 54, 754-775	1.3	138
88	Formalizing requirements with object models and temporal constraints. *Software and Systems Modeling*, **2011**, 10, 147-160	1.9	14

87	HyDI: A Language for Symbolic Hybrid Systems with Discrete Interaction **2011**,		12
86	OthelloPlay **2011**,		2
85	Boosting Lazy Abstraction for SystemC with Partial Order Reduction. *Lecture Notes in Computer Science*, **2011**, 341-356	0.9	11
84	Kratos A Software Model Checker for SystemC. *Lecture Notes in Computer Science*, **2011**, 310-316	0.9	32
83	An Analytic Evaluation of SystemC Encodings in Promela. *Lecture Notes in Computer Science*, **2011**, 90-107	0.9	6
82	Efficient Scenario Verification for Hybrid Automata. *Lecture Notes in Computer Science*, **2011**, 317-332	0.9	2
81	From Sequential Extended Regular Expressions to NFA with Symbolic Labels. *Lecture Notes in Computer Science*, **2011**, 87-94	0.9	
80	Efficient generation of craig interpolants in satisfiability modulo theories. *ACM Transactions on Computational Logic*, **2010**, 12, 1-54	0.9	38
79	SMT-Based Software Model Checking. *Lecture Notes in Computer Science*, **2010**, 1-3	0.9	1
78	Parametric analysis of distributed firm real-time systems: A case study **2010**,		2
77	Satisfiability Modulo the Theory of Costs: Foundations and Applications. *Lecture Notes in Computer Science*, **2010**, 99-113	0.9	47
76	Model Checking of Hybrid Systems Using Shallow Synchronization. *Lecture Notes in Computer Science*, **2010**, 155-169	0.9	7
75	RATSY A New Requirements Analysis Tool with Synthesis. *Lecture Notes in Computer Science*, **2010** , 425-429	0.9	60
74	A Model Checker for AADL. *Lecture Notes in Computer Science*, **2010**, 562-565	0.9	16
73	Verification and performance evaluation of aadl models **2009**,		6
72	Delayed theory combination vs. Nelson-Oppen for satisfiability modulo theories: a comparative analysis. *Annals of Mathematics and Artificial Intelligence*, **2009**, 55, 63-99	0.8	10
71	Structure-aware computation of predicate abstraction **2009**,		5
70	**2009**,		83

69	Supporting Requirements Validation: The EuRailCheck Tool **2009**,		6
68	Codesign of dependable systems: A component-based modeling language **2009**,		6
67	Requirements Validation for Hybrid Systems. *Lecture Notes in Computer Science*, **2009**, 188-203	0.9	30
66	Interpolant Generation for UTVPI. *Lecture Notes in Computer Science*, **2009**, 167-182	0.9	9
65	From Informal Requirements to Property-Driven Formal Validation. *Lecture Notes in Computer Science*, **2009**, 166-181	0.9	7
64	The COMPASS Approach: Correctness, Modelling and Performability of Aerospace Systems. *Lecture Notes in Computer Science*, **2009**, 173-186	0.9	44
63	Symbolic Compilation of PSL. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **2008**, 27, 1737-1750	2.5	13
62	Symbolic Computation of Schedulability Regions Using Parametric Timed Automata **2008**,		30
61	Beyond Boolean SAT: Satisfiability modulo theories **2008**,		6
60	Object Models with Temporal Constraints **2008**,		3
59	Chapter 22 Automated Planning. *Foundations of Artificial Intelligence*, **2008**, 841-867		6
58	The MathSAT 4 SMT Solver. *Lecture Notes in Computer Science*, **2008**, 299-303	0.9	93
57	Efficient Interpolant Generation in Satisfiability Modulo Theories **2008**, 397-412		43
56	Computing Predicate Abstractions by Integrating BDDs and SMT Solvers **2007**,		22
55	SYMBOLIC IMPLEMENTATION OF ALTERNATING AUTOMATA. *International Journal of Foundations of Computer Science*, **2007**, 18, 727-743	0.6	9
54	Syntactic Optimizations for PSL Verification **2007**, 505-518		5
53	A Simple and Flexible Way of Computing Small Unsatisfiable Cores in SAT Modulo Theories **2007**, 334-339		20
52	Boolean Abstraction for Temporal Logic Satisfiability. *Lecture Notes in Computer Science*, **2007**, 532-546	0.9	25

51	A Lazy and Layered SMT((mathcal{BV})) Solver for Hard Industrial Verification Problems **2007**, 547-560		22
50	Symbolic Fault Tree Analysis for Reactive Systems **2007**, 162-176		34
49	Verifying Heap-Manipulating Programs in an SMT Framework **2007**, 237-252		9
48	Efficient theory combination via boolean search. *Information and Computation*, **2006**, 204, 1493-1525	0.8	38
47	**2006**,		15
46	Encoding RTL Constructs for MathSAT: a Preliminary Report. *Electronic Notes in Theoretical Computer Science*, **2006**, 144, 3-14	0.7	13
45	Strong planning under partial observability. *Artificial Intelligence*, **2006**, 170, 337-384	3.6	31
44	Delayed Theory Combination vs. Nelson-Oppen for Satisfiability Modulo Theories: A Comparative Analysis. *Lecture Notes in Computer Science*, **2006**, 527-541	0.9	5
43	A Symbolic Model Checking Framework for Safety Analysis, Diagnosis, and Synthesis. *Lecture Notes in Computer Science*, **2006**, 1-18	0.9	8
42	Building Efficient Decision Procedures on Top of SAT Solvers. *Lecture Notes in Computer Science*, **2006**, 144-175	0.9	5
41	To Ackermann-ize or Not to Ackermann-ize? On Efficiently Handling Uninterpreted Function Symbols in (mathit{SMT}(mathcal{EUF} cup mathcal{T})). *Lecture Notes in Computer Science*, **2006**, 557-571	0.9	9
40	An Incremental and Layered Procedure for the Satisfiability of Linear Arithmetic Logic. *Lecture Notes in Computer Science*, **2005**, 317-333	0.9	34
39	Verifying Industrial Hybrid Systems with MathSAT. *Electronic Notes in Theoretical Computer Science*, **2005**, 119, 17-32	0.7	46
38	MathSAT: Tight Integration of SAT and Mathematical Decision Procedures. *Journal of Automated Reasoning*, **2005**, 35, 265-293	1	41
37	Efficient Satisfiability Modulo Theories via Delayed Theory Combination. *Lecture Notes in Computer Science*, **2005**, 335-349	0.9	26
36	The MathSAT 3 System. *Lecture Notes in Computer Science*, **2005**, 315-321	0.9	28
35	MathSAT: Tight Integration of SAT and Mathematical Decision Procedures **2005**, 265-293		4
34	Bounded Verification of Past LTL. *Lecture Notes in Computer Science*, **2004**, 245-259	0.9	15

33	Bounded Model Checking. *Advances in Computers*, **2003**, 58, 117-148	2.9	332
32	Validation of Multiagent Systems by Symbolic Model Checking. *Lecture Notes in Computer Science*, **2003**, 32-46	0.9	4
31	Bounded Model Checking for Past LTL. *Lecture Notes in Computer Science*, **2003**, 18-33	0.9	27
30	NuSMV 2: An OpenSource Tool for Symbolic Model Checking. *Lecture Notes in Computer Science*, **2002**, 359-364	0.9	686
29	A SAT Based Approach for Solving Formulas over Boolean and Linear Mathematical Propositions. *Lecture Notes in Computer Science*, **2002**, 195-210	0.9	48
28	Integrating Boolean and Mathematical Solving: Foundations, Basic Algorithms, and Requirements. *Lecture Notes in Computer Science*, **2002**, 231-245	0.9	4
27	Integrating BDD-Based and SAT-Based Symbolic Model Checking. *Lecture Notes in Computer Science* , **2002**, 49-56	0.9	9
26	Improving the Encoding of LTL Model Checking into SAT. *Lecture Notes in Computer Science*, **2002**, 196-207	0.9	13
25	Searching Powerset Automata by Combining Explicit-State and Symbolic Model Checking. *Lecture Notes in Computer Science*, **2001**, 313-327	0.9	7
24	Industrial Applications of Model Checking. *Lecture Notes in Computer Science*, **2001**, 153-168	0.9	8
23	Verification of a safety-critical railway interlocking system with real-time constraints. *Science of Computer Programming*, **2000**, 36, 53-64	1.1	17
22	NUSMV: a new symbolic model checker. *International Journal on Software Tools for Technology Transfer*, **2000**, 2, 410-425	1.3	317
21	A Context-Based Mechanization of Multi-Agent Reasoning. *Applied Logic Series*, **2000**, 65-83		
20	Conformant Planning via Model Checking. *Lecture Notes in Computer Science*, **2000**, 21-34	0.9	11
19	Symbolic Model Checking without BDDs. *Lecture Notes in Computer Science*, **1999**, 193-207	0.9	654
18	Symbolic Model Checking without BDDs **1999**,		237
17	A Many-Sorted Natural Deduction. *Computational Intelligence*, **1998**, 14, 134-149	2.5	1
16	Formal Verification of a Railway Interlocking System using Model Checking. *Formal Aspects of Computing*, **1998**, 10, 361-380	1.2	43

List of Publications

15	Model Checking Safety Critical Software with SPIN: an Application to a Railway Interlocking System. *Lecture Notes in Computer Science*, **1998**, 284-293	0.9	12
14	A Structured Approach to the Formal Certification of Safety of Computer Aided Development Tools. *Lecture Notes in Computer Science*, **1998**, 221-230	0.9	
13	Formal specification of beliefs in multi-agent systems. *Lecture Notes in Computer Science*, **1997**, 117-130	0.9	1
12	A provably correct embedded verifier for the certification of safety critical software. *Lecture Notes in Computer Science*, **1997**, 202-213	0.9	9
11	Planning via model checking: A decision procedure for AR. *Lecture Notes in Computer Science*, **1997**, 130-142	0.9	31
10	Mechanizing multi-agent reasoning with belief contexts. *Lecture Notes in Computer Science*, **1996**, 694-696	0.9	2
9	Flexible planning by integrating multilevel reasoning. *Engineering Applications of Artificial Intelligence*, **1995**, 8, 401-412	7.2	1
8	Multi-agent reasoning with belief contexts: the approach and a case study. *Lecture Notes in Computer Science*, **1995**, 71-85	0.9	15
7	MRG: BUILDING PLANNERS FOR REAL-WORLD COMPLEX APPLICATIONS. *Applied Artificial Intelligence*, **1994**, 8, 333-357	2.3	2
6	Introspective metatheoretic reasoning. *Lecture Notes in Computer Science*, **1994**, 425-439	0.9	2
5	Building and executing proof strategies in a formal metatheory. *Lecture Notes in Computer Science*, **1993**, 11-22	0.9	1
4	Extended bounded response LTL: a new safety fragment for efficient reactive synthesis. *Formal Methods in System Design*,1	1.4	
3	Requirements Refinement and Component Reuse1397-1432		
2	Formalization and Validation of Safety-Critical Requirements. *Electronic Proceedings in Theoretical Computer Science, EPTCS*,20, 68-75		1
1	Temporal logic satisfiability for the design of complex systems. *Electronic Proceedings in Theoretical Computer Science, EPTCS*,119, 4-6		