

Javier Esparza

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/2328530/publications.pdf>

Version: 2024-02-01

158
papers

4,536
citations

172207

29
h-index

161609

54
g-index

172
all docs

172
docs citations

172
times ranked

1079
citing authors

#	ARTICLE	IF	CITATIONS
1	Finding Cut-Offs in Leaderless Rendez-Vous Protocols is Easy. Lecture Notes in Computer Science, 2021, , 42-61.	1.0	1
2	Computing Parameterized Invariants of Parameterized Petri Nets. Lecture Notes in Computer Science, 2021, , 141-163.	1.0	1
3	Back to the Future: A Fresh Look at Linear Temporal Logic. Lecture Notes in Computer Science, 2021, , 3-13.	1.0	1
4	The complexity of verifying population protocols. Distributed Computing, 2021, 34, 133-177.	0.7	5
5	Towards efficient verification of population protocols. Formal Methods in System Design, 2021, 57, 305-342.	0.9	1
6	Lower Bounds on the State Complexity of Population Protocols. , 2021, , .		5
7	Population Protocols: Beyond Runtime Analysis. Lecture Notes in Computer Science, 2021, , 28-51.	1.0	0
8	Structural Invariants for the Verification of Systems with Parameterized Architectures. Lecture Notes in Computer Science, 2020, , 228-246.	1.0	7
9	Checking Qualitative Liveness Properties of Replicated Systems with Stochastic Scheduling. Lecture Notes in Computer Science, 2020, , 372-397.	1.0	7
10	Complexity of Verification and Synthesis of Threshold Automata. Lecture Notes in Computer Science, 2020, , 144-160.	1.0	5
11	An Efficient Normalisation Procedure for Linear Temporal Logic and Very Weak Alternating Automata. , 2020, , .		7
12	A Unified Translation of Linear Temporal Logic to M^{A} -Automata. Journal of the ACM, 2020, 67, 1-61.	1.8	7
13	Negotiation as concurrency primitive. Acta Informatica, 2019, 56, 93-159.	0.5	3
14	Parameterized Analysis of Immediate Observation Petri Nets. Lecture Notes in Computer Science, 2019, , 365-385.	1.0	8
15	Computing the Expected Execution Time of Probabilistic Workflow Nets. Lecture Notes in Computer Science, 2019, , 154-171.	1.0	4
16	Preface for the special issue GandALF 2015. Acta Informatica, 2018, 55, 89-90.	0.5	0
17	One Theorem to Rule Them All. , 2018, , .		22
18	Black Ninjas in the Dark. , 2018, , .		3

#	ARTICLE	IF	CITATIONS
19	Model Checking Procedural Programs. , 2018, , 541-572.		11
20	Peregrine: A Tool for the Analysis of Population Protocols. Lecture Notes in Computer Science, 2018, , 604-611.	1.0	9
21	Computing the Concurrency Threshold of Sound Free-Choice Workflow Nets. Lecture Notes in Computer Science, 2018, , 3-19.	1.0	2
22	Advances in Parameterized Verification of Population Protocols. Lecture Notes in Computer Science, 2017, , 7-14.	1.0	1
23	Model checking parameterized asynchronous shared-memory systems. Formal Methods in System Design, 2017, 50, 140-167.	0.9	3
24	Minimizing Test Suites with Unfoldings of Multithreaded Programs. Transactions on Embedded Computing Systems, 2017, 16, 1-24.	2.1	7
25	Towards Efficient Verification of Population Protocols. , 2017, , .		7
26	Polynomial analysis algorithms for free choice Probabilistic Workflow Nets. Performance Evaluation, 2017, 117, 104-129.	0.9	8
27	Verification of population protocols. Acta Informatica, 2017, 54, 191-215.	0.5	27
28	Static analysis of deterministic negotiations. , 2017, , .		4
29	From LTL and Limit-Deterministic Büchi Automata to Deterministic Parity Automata. Lecture Notes in Computer Science, 2017, , 426-442.	1.0	25
30	Existence of home states in Petri nets is decidable. Information Processing Letters, 2016, 116, 423-427.	0.4	14
31	From LTL to deterministic automata. Formal Methods in System Design, 2016, 49, 219-271.	0.9	20
32	Parameterized Verification of Asynchronous Shared-Memory Systems. Journal of the ACM, 2016, 63, 1-48.	1.8	15
33	Limit-Deterministic Büchi Automata for Linear Temporal Logic. Lecture Notes in Computer Science, 2016, , 312-332.	1.0	42
34	Polynomial Analysis Algorithms for Free Choice Probabilistic Workflow Nets. Lecture Notes in Computer Science, 2016, , 89-104.	1.0	4
35	Reduction Rules for Colored Workflow Nets. Lecture Notes in Computer Science, 2016, , 342-358.	1.0	14
36	Negotiations and Petri Nets. Lecture Notes in Computer Science, 2016, , 203-225.	1.0	3

#	ARTICLE	IF	CITATIONS
37	Unfolding Based Minimal Test Suites for Testing Multithreaded Programs. , 2015, , .		3
38	An SMT-based approach to fair termination analysis. , 2015, , .		5
39	Distributed Markov Chains. Lecture Notes in Computer Science, 2015, , 117-134.	1.0	3
40	Model Checking Parameterized Asynchronous Shared-Memory Systems. Lecture Notes in Computer Science, 2015, , 67-84.	1.0	10
41	Negotiation Programs. Lecture Notes in Computer Science, 2015, , 157-178.	1.0	1
42	FPSOLVE: A Generic Solver for Fixpoint Equations Over Semirings. International Journal of Foundations of Computer Science, 2015, 26, 805-825.	0.8	1
43	Pattern-Based Verification for Multithreaded Programs. ACM Transactions on Programming Languages and Systems, 2014, 36, 1-29.	1.7	15
44	From LTL to Deterministic Automata: A Safrless Compositional Approach. Lecture Notes in Computer Science, 2014, , 192-208.	1.0	33
45	An SMT-Based Approach to Coverability Analysis. Lecture Notes in Computer Science, 2014, , 603-619.	1.0	47
46	On Negotiation as Concurrency Primitive II: Deterministic Cyclic Negotiations. Lecture Notes in Computer Science, 2014, , 258-273.	1.0	9
47	Deterministic Negotiations: Concurrency for Free. Lecture Notes in Computer Science, 2014, , 23-31.	1.0	0
48	Message-Passing Algorithms for the Verification of Distributed Protocols. Lecture Notes in Computer Science, 2014, , 222-241.	1.0	0
49	Analyzing probabilistic pushdown automata. Formal Methods in System Design, 2013, 43, 124-163.	0.9	28
50	A strongly polynomial algorithm for criticality of branching processes and consistency of stochastic context-free grammars. Information Processing Letters, 2013, 113, 381-385.	0.4	3
51	A Fully Verified Executable LTL Model Checker. Lecture Notes in Computer Science, 2013, , 463-478.	1.0	55
52	Parameterized Verification of Asynchronous Shared-Memory Systems. Lecture Notes in Computer Science, 2013, , 124-140.	1.0	21
53	On Negotiation as Concurrency Primitive. Lecture Notes in Computer Science, 2013, , 440-454.	1.0	12
54	Reactive and Proactive Diagnosis of Distributed Systems Using Net Unfoldings. , 2012, , .		1

#	ARTICLE	IF	CITATIONS
55	A Perfect Model for Bounded Verification. , 2012, , .		14
56	Space-efficient scheduling of stochastically generated tasks. Information and Computation, 2012, 210, 87-110.	0.5	2
57	Proving Termination of Probabilistic Programs Using Patterns. Lecture Notes in Computer Science, 2012, , 123-138.	1.0	37
58	Deterministic Automata for the (F,G)-Fragment of LTL. Lecture Notes in Computer Science, 2012, , 7-22.	1.0	36
59	Rabinizer: Small Deterministic Automata for LTL(F,G). Lecture Notes in Computer Science, 2012, , 72-76.	1.0	16
60	Learning Workflow Petri Nets. Fundamenta Informaticae, 2011, 113, 205-228.	0.3	5
61	Parikh's theorem: A simple and direct automaton construction. Information Processing Letters, 2011, 111, 614-619.	0.4	43
62	Derivation tree analysis for accelerated fixed-point computation. Theoretical Computer Science, 2011, 412, 3226-3241.	0.5	3
63	Complexity of pattern-based verification for multithreaded programs. , 2011, , .		16
64	Complexity of pattern-based verification for multithreaded programs. ACM SIGPLAN Notices, 2011, 46, 499-510.	0.2	13
65	Probabilistic Abstractions with Arbitrary Domains. Lecture Notes in Computer Science, 2011, , 334-350.	1.0	5
66	On least fixed points of systems of positive polynomials. ACM Communications in Computer Algebra, 2010, 43, 81-83.	0.2	0
67	Newtonian program analysis. Journal of the ACM, 2010, 57, 1-47.	1.8	39
68	Computing the Least Fixed Point of Positive Polynomial Systems. SIAM Journal on Computing, 2010, 39, 2282-2335.	0.8	19
69	Learning Workflow Petri Nets. Lecture Notes in Computer Science, 2010, , 206-225.	1.0	7
70	Automatic Error Correction of Java Programs. Lecture Notes in Computer Science, 2010, , 67-81.	1.0	7
71	Verification of Graph Transformation Systems with Context-Free Specifications. Lecture Notes in Computer Science, 2010, , 107-122.	1.0	8
72	A False History of True Concurrency: From Petri to Tools. Lecture Notes in Computer Science, 2010, , 180-186.	1.0	4

#	ARTICLE	IF	CITATIONS
73	Analysis of Systems with Stochastic Process Creation. Lecture Notes in Computer Science, 2010, , 1-1.	1.0	0
74	Modeling and Verification for Timing Satisfaction of Fault-Tolerant Systems with Finiteness. , 2009, , .		0
75	Examining robotic systems with shape-adjustable manipulators under dynamic environments: From simulation to verification. , 2009, , .		0
76	Stochastic Process Creation. Lecture Notes in Computer Science, 2009, , 24-33.	1.0	0
77	A negative result on depth-first net unfoldings. International Journal on Software Tools for Technology Transfer, 2008, 10, 161-166.	1.7	4
78	Approximative Methods for Monotone Systems of Min-Max-Polynomial Equations. Lecture Notes in Computer Science, 2008, , 698-710.	1.0	13
79	Newton's Method for ω -Continuous Semirings. Lecture Notes in Computer Science, 2008, , 14-26.	1.0	11
80	SDSIrep: A Reputation System Based on SDSI. , 2008, , 501-516.		4
81	Symbolic Context-Bounded Analysis of Multithreaded Java Programs. Lecture Notes in Computer Science, 2008, , 270-287.	1.0	34
82	Solving Monotone Polynomial Equations. International Federation for Information Processing, 2008, , 285-298.	0.4	2
83	On the convergence of Newton's method for monotone systems of polynomial equations. , 2007, , .		26
84	On Fixed Point Equations over Commutative Semirings. , 2007, , 296-307.		22
85	jMoped: A Test Environment for Java Programs. , 2007, , 164-167.		9
86	An Extension of Newton's Method to ω -Continuous Semirings. Lecture Notes in Computer Science, 2007, , 157-168.	1.0	14
87	Abstraction Refinement with Craig Interpolation and Symbolic Pushdown Systems. Lecture Notes in Computer Science, 2006, , 489-503.	1.0	31
88	Rewriting Models of Boolean Programs. Lecture Notes in Computer Science, 2006, , 136-150.	1.0	12
89	Efficient Algorithms for Alternating Pushdown Systems with an Application to the Computation of Certificate Chains. Lecture Notes in Computer Science, 2006, , 141-153.	1.0	18
90	Separability in Conflict-Free Petri Nets. , 2006, , 1-18.		7

#	ARTICLE	IF	CITATIONS
91	Model Checking Probabilistic Pushdown Automata. Logical Methods in Computer Science, 2006, 2, .	0.4	24
92	Monotonic Set-Extended Prefix Rewriting and Verification of Recursive Ping-Pong Protocols. Lecture Notes in Computer Science, 2006, , 415-429.	1.0	0
93	Reachability Analysis of Synchronized PA Systems. Electronic Notes in Theoretical Computer Science, 2005, 138, 153-178.	0.9	8
94	A Note on On-the-Fly Verification Algorithms. Lecture Notes in Computer Science, 2005, , 174-190.	1.0	81
95	jMoped: A Java Bytecode Checker Based on Moped. Lecture Notes in Computer Science, 2005, , 541-545.	1.0	22
96	Locality-Based Abstractions. Lecture Notes in Computer Science, 2005, , 118-134.	1.0	4
97	Verifying Probabilistic Procedural Programs. Lecture Notes in Computer Science, 2004, , 16-31.	1.0	15
98	Model checking LTL with regular valuations for pushdown systems. Information and Computation, 2003, 186, 355-376.	0.5	107
99	A Generic Approach to the Static Analysis of Concurrent Programs with Procedures. International Journal of Foundations of Computer Science, 2003, 14, 551-582.	0.8	23
100	The Model-Checking Kit. Lecture Notes in Computer Science, 2003, , 463-472.	1.0	23
101	An Automata-Theoretic Approach to Software Verification. Lecture Notes in Computer Science, 2003, , 21-21.	1.0	3
102	A generic approach to the static analysis of concurrent programs with procedures. , 2003, , .		90
103	A generic approach to the static analysis of concurrent programs with procedures. ACM SIGPLAN Notices, 2003, 38, 62-73.	0.2	19
104	Simple Representative Instantiations for Multicast Protocols. Lecture Notes in Computer Science, 2003, , 128-143.	1.0	2
105	An Improvement of McMillan's Unfolding Algorithm. Formal Methods in System Design, 2002, 20, 285-310.	0.9	191
106	Grammars as Processes. Lecture Notes in Computer Science, 2002, , 277-297.	1.0	11
107	An Algebraic Approach to the Static Analysis of Concurrent Software. Lecture Notes in Computer Science, 2002, , 3-3.	1.0	0
108	A BDD-Based Model Checker for Recursive Programs. Lecture Notes in Computer Science, 2001, , 324-336.	1.0	92

#	ARTICLE	IF	CITATIONS
109	Implementing LTL model checking with net unfoldings. Lecture Notes in Computer Science, 2001, , 37-56.	1.0	25
110	Model-Checking LTL with Regular Valuations for Pushdown Systems. Lecture Notes in Computer Science, 2001, , 316-339.	1.0	24
111	Verification of Systems with an Infinite State Space. Lecture Notes in Computer Science, 2001, , 183-186.	1.0	3
112	Model checking (with) declarative programs. , 2001, , .		0
113	An efficient automata approach to some problems on context-free grammars. Information Processing Letters, 2000, 74, 221-227.	0.4	19
114	Verification of Safety Properties Using Integer Programming: Beyond the State Equation. Formal Methods in System Design, 2000, 16, 159-189.	0.9	41
115	Efficient algorithms for pre* and post* on interprocedural parallel flow graphs. , 2000, , .		59
116	Efficient Algorithms for Model Checking Pushdown Systems. Lecture Notes in Computer Science, 2000, , 232-247.	1.0	211
117	Verifying Single and Multi-mutator Garbage Collectors with Owicki-Gries in Isabelle/HOL. Lecture Notes in Computer Science, 2000, , 619-628.	1.0	8
118	A New Unfolding Approach to LTL Model Checking. Lecture Notes in Computer Science, 2000, , 475-486.	1.0	19
119	Petri Nets and Regular Processes. Journal of Computer and System Sciences, 1999, 59, 476-503.	0.9	26
120	A Logical Viewpoint on Process-Algebraic Quotients. Lecture Notes in Computer Science, 1999, , 499-514.	1.0	4
121	Constraint-Based Analysis of Broadcast Protocols. Lecture Notes in Computer Science, 1999, , 50-66.	1.0	31
122	An Unfolding Algorithm for Synchronous Products of Transition Systems. Lecture Notes in Computer Science, 1999, , 2-20.	1.0	34
123	Proof-Checking Protocols Using Bisimulations. Lecture Notes in Computer Science, 1999, , 525-540.	1.0	4
124	An Automata-Theoretic Approach to Interprocedural Data-Flow Analysis. Lecture Notes in Computer Science, 1999, , 14-30.	1.0	59
125	Reachability in live and safe free-choice Petri nets is NP-complete. Theoretical Computer Science, 1998, 198, 211-224.	0.5	29
126	Decidability and complexity of Petri net problems " An introduction. Lecture Notes in Computer Science, 1998, , 374-428.	1.0	146

#	ARTICLE	IF	CITATIONS
127	Petri Nets, Commutative Context-Free Grammars, and Basic Parallel Processes. <i>Fundamenta Informaticae</i> , 1997, 31, 13-25.	0.3	77
128	Decidability of model checking for infinite-state concurrent systems. <i>Acta Informatica</i> , 1997, 34, 85-107.	0.5	114
129	Reachability analysis of pushdown automata: Application to model-checking. <i>Lecture Notes in Computer Science</i> , 1997, , 135-150.	1.0	352
130	Verification using PEP. <i>Lecture Notes in Computer Science</i> , 1996, , 591-594.	1.0	3
131	Trapping mutual exclusion in the box calculus. <i>Theoretical Computer Science</i> , 1996, 153, 95-128.	0.5	16
132	An improvement of McMillan's unfolding algorithm. <i>Lecture Notes in Computer Science</i> , 1996, , 87-106.	1.0	100
133	Checking system properties via integer programming. <i>Lecture Notes in Computer Science</i> , 1996, , 250-264.	1.0	17
134	An effective tableau system for the linear time $\hat{1}4$ -calculus. <i>Lecture Notes in Computer Science</i> , 1996, , 98-109.	1.0	7
135	Deciding finiteness of Petri nets up to bisimulation. <i>Lecture Notes in Computer Science</i> , 1996, , 478-489.	1.0	26
136	Complexity results for 1-safe nets. <i>Theoretical Computer Science</i> , 1995, 147, 117-136.	0.5	94
137	Petri nets, commutative context-free grammars, and basic parallel processes. <i>Lecture Notes in Computer Science</i> , 1995, , 221-232.	1.0	28
138	Model checking using net unfoldings. <i>Science of Computer Programming</i> , 1994, 23, 151-195.	1.5	140
139	On the decidability of model checking for several $\hat{1}4$ -calculi and Petri nets. , 1994, , 115-129.		35
140	Operational Semantics for the Petri Box Calculus. <i>Lecture Notes in Computer Science</i> , 1994, , 210-225.	1.0	19
141	Decidability Issues for Petri Nets. <i>BRICS Report Series</i> , 1994, 1, .	0.2	46
142	The asynchronous committee meeting problem. <i>Lecture Notes in Computer Science</i> , 1994, , 276-287.	1.0	0
143	Reachability in cyclic extended free-choice systems. <i>Theoretical Computer Science</i> , 1993, 114, 93-118.	0.5	32
144	General refinement and recursion operators for the Petri Box calculus. <i>Lecture Notes in Computer Science</i> , 1993, , 130-140.	1.0	38

#	ARTICLE	IF	CITATIONS
145	Model checking using net unfoldings. Lecture Notes in Computer Science, 1993, , 613-628.	1.0	26
146	Complexity results for 1-safe nets. Lecture Notes in Computer Science, 1993, , 326-337.	1.0	28
147	Complexity Results for 1-safe Nets. DAIMI Report Series, 1993, 22, .	0.1	6
148	Traps characterize home states in free choice systems. Theoretical Computer Science, 1992, 101, 161-176.	0.5	21
149	A polynomial-time algorithm to decide liveness of bounded free choice nets. Theoretical Computer Science, 1992, 102, 185-205.	0.5	57
150	A solution to the covering problem for 1-bounded conflict-free Petri nets using Linear Programming. Information Processing Letters, 1992, 41, 313-319.	0.4	11
151	Reachability in reversible Free Choice systems. , 1991, , 384-397.		9
152	Zeros of the Hankel function of real order out of the principal Riemann sheet. Journal of Computational and Applied Mathematics, 1991, 37, 89-99.	1.1	4
153	Model checking of persistent Petri nets. , 1991, , 35-52.		9
154	Compositional synthesis of live and bounded free choice Petri nets. Lecture Notes in Computer Science, 1991, , 172-187.	1.0	12
155	Synthesis rules for Petri nets, and how they lead to new results. , 1990, , 182-198.		17
156	Top-down synthesis of live and bounded free choice nets. , 1990, , 118-139.		21
157	Abduction of trap invariants in parameterized systems. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 346, 1-17.	0.8	0
158	Derivation Tree Analysis for Accelerated Fixed-Point Computation. Lecture Notes in Computer Science, 0, , 301-313.	1.0	3