Samir Genaim

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/5830001/publications.pdf

Version: 2024-02-01

78	1,363	430754	434063
papers	citations	18	31
		h-index	g-index

82	82	82	350
all docs	docs citations	times ranked	citing authors

#	Article	IF	Citations
1	Lower-Bound Synthesis Using Loop Specialization and Max-SMT. Lecture Notes in Computer Science, 2021, , 863-886.	1.0	4
2	A Transformational Approach to Resource Analysis with Typed-norms Inference. Theory and Practice of Logic Programming, 2020, 20, 310-357.	1.1	3
3	Control-Flow Refinement by Partial Evaluation, and its Application to Termination and Cost Analysis. Theory and Practice of Logic Programming, 2019, 19, 990-1005.	1.1	12
4	Multiphase-Linear Ranking Functions and Their Relation to Recurrent Sets. Lecture Notes in Computer Science, 2019, , 459-480.	1.0	19
5	Rely-Guarantee Termination and Cost Analyses of Loops with Concurrent Interleavings. Journal of Automated Reasoning, 2017, 59, 47-85.	1.1	5
6	On Multiphase-Linear Ranking Functions. Lecture Notes in Computer Science, 2017, , 601-620.	1.0	30
7	EasyInterface: A Toolkit for Rapid Development of GUIs for Research Prototype Tools. Lecture Notes in Computer Science, 2017, , 379-383.	1.0	4
8	May-Happen-in-Parallel Analysis with Returned Futures. Lecture Notes in Computer Science, 2017, , 42-58.	1.0	1
9	May-Happen-in-Parallel Analysis for Actor-Based Concurrency. ACM Transactions on Computational Logic, 2016, 17, 1-39.	0.7	15
10	A formal verification framework for static analysis. Software and Systems Modeling, 2016, 15, 987-1012.	2.2	9
11	May-Happen-in-Parallel Analysis with Condition Synchronization. Lecture Notes in Computer Science, 2016, , 1-19.	1.0	0
12	Objectâ€sensitive cost analysis for concurrent objects. Software Testing Verification and Reliability, 2015, 25, 218-271.	1.7	19
13	From non-zenoness verification to termination. , 2015, , .		0
14	Preparing HPC Applications for Exascale: Challenges and Recommendations. , 2015, , .		19
15	A practical comparator of cost functions and its applications. Science of Computer Programming, 2015, 111, 483-504.	1.5	3
16	Resource Analysis: From Sequential to Concurrent and Distributed Programs. Lecture Notes in Computer Science, 2015, , 3-17.	1.0	3
17	Complexity of Bradley-Manna-Sipma Lexicographic Ranking Functions. Lecture Notes in Computer Science, 2015, , 304-321.	1.0	7
18	May-Happen-in-Parallel Analysis for Asynchronous Programs with Inter-Procedural Synchronization. Lecture Notes in Computer Science, 2015, , 72-89.	1.0	7

#	Article	IF	Citations
19	Inference of Field-Sensitive Reachability and Cyclicity. ACM Transactions on Computational Logic, 2014, 15, 1-41.	0.7	2
20	Ranking Functions for Linear-Constraint Loops. Journal of the ACM, 2014, 61, 1-55.	1.8	56
21	Conditional termination of loops over heap-allocated data. Science of Computer Programming, 2014, 92, 2-24.	1.5	5
22	A Transformational Approach to Resource Analysis with Typed-Norms. Lecture Notes in Computer Science, 2014, , 38-53.	1.0	2
23	SACO: Static Analyzer for Concurrent Objects. Lecture Notes in Computer Science, 2014, , 562-567.	1.0	38
24	Corrigendum to â€œReachability-based acyclicity analysis by abstract interpretationâ€•[Theoretical Computer Science 474 (2013) 60â€“79]. Theoretical Computer Science, 2013, 503, 115.	0.5	0
25	Heap space analysis for garbage collected languages. Science of Computer Programming, 2013, 78, 1427-1448.	1.5	9
26	Reachability-based acyclicity analysis by Abstract Interpretation. Theoretical Computer Science, 2013, 474, 60-79.	0.5	7
27	On the Inference of Resource Usage Upper and Lower Bounds. ACM Transactions on Computational Logic, 2013, 14, 1-35.	0.7	29
28	On the linear ranking problem for integer linear-constraint loops. , 2013, , .		32
29	Precise Cost Analysis via Local Reasoning. Lecture Notes in Computer Science, 2013, , 319-333.	1.0	5
30	Termination and Cost Analysis of Loops with Concurrent Interleavings. Lecture Notes in Computer Science, 2013, , 349-364.	1.0	15
31	May-Happen-in-Parallel Based Deadlock Analysis for Concurrent Objects. Lecture Notes in Computer Science, 2013, , 273-288.	1.0	30
32	Proving Termination Starting from the End. Lecture Notes in Computer Science, 2013, , 397-412.	1.0	18
33	On the linear ranking problem for integer linear-constraint loops. ACM SIGPLAN Notices, 2013, 48, 51-62.	0.2	15
34	May-Happen-in-Parallel Analysis for Priority-Based Scheduling. Lecture Notes in Computer Science, 2013, , 18-34.	1.0	1
35	Automatic Inference of Bounds on Resource Consumption. Lecture Notes in Computer Science, 2013, , 119-144.	1.0	2
36	On the Termination of Integer Loops. ACM Transactions on Programming Languages and Systems, 2012, 34, 1-24.	1.7	18

#	Article	IF	Citations
37	MayPar. , 2012, , .		2
38	COSTABS. , 2012, , .		22
39	Cost analysis of object-oriented bytecode programs. Theoretical Computer Science, 2012, 413, 142-159.	0.5	96
40	Automatic Inference of Resource Consumption Bounds. Lecture Notes in Computer Science, 2012, , 1-11.	1.0	10
41	Verified Resource Guarantees for Heap Manipulating Programs. Lecture Notes in Computer Science, 2012, , 130-145.	1.0	9
42	Analysis of May-Happen-in-Parallel in Concurrent Objects. Lecture Notes in Computer Science, 2012, , 35-51.	1.0	24
43	On the Limits of the Classical Approach to Cost Analysis. Lecture Notes in Computer Science, 2012, , 405-421.	1.0	34
44	On the Termination of Integer Loops. Lecture Notes in Computer Science, 2012, , 72-87.	1.0	9
45	Handling Non-linear Operations in the Value Analysis of COSTA. Electronic Notes in Theoretical Computer Science, 2011, 279, 3-17.	0.9	5
46	Closed-Form Upper Bounds in Static Cost Analysis. Journal of Automated Reasoning, 2011, 46, 161-203.	1.1	121
47	Verified resource guarantees using COSTA and KeY. , 2011, , .		17
48	Task-level analysis for a language with async/finish parallelism. , 2011, , .		9
49	Task-level analysis for a language with async/finish parallelism. ACM SIGPLAN Notices, 2011, 46, 21-30.	0.2	2
50	Simulating Concurrent Behaviors with Worst-Case Cost Bounds. Lecture Notes in Computer Science, 2011, , 353-368.	1.0	17
51	Cost Analysis of Concurrent OO Programs. Lecture Notes in Computer Science, 2011, , 238-254.	1.0	29
52	Parametric inference of memory requirements for garbage collected languages. , 2010, , .		29
53	Comparing Cost Functions in Resource Analysis. Lecture Notes in Computer Science, 2010, , 1-17.	1.0	7
54	From Object Fields to Local Variables: A Practical Approach to Field-Sensitive Analysis. Lecture Notes in Computer Science, 2010, , 100-116.	1.0	7

#	Article	IF	Citations
55	Parametric inference of memory requirements for garbage collected languages. ACM SIGPLAN Notices, 2010, 45, 121-130.	0.2	3
56	A declarative encoding of telecommunications feature subscription in SAT. , 2009, , .		4
57	Cost Relation Systems: A Language-Independent Target Language for Cost Analysis. Electronic Notes in Theoretical Computer Science, 2009, 248, 31-46.	0.9	4
58	Termination and Cost Analysis with COSTA and its User Interfaces. Electronic Notes in Theoretical Computer Science, 2009, 258, 109-121.	0.9	25
59	Live heap space analysis for languages with garbage collection. , 2009, , .		31
60	Resource Usage Analysis and Its Application to Resource Certification. Lecture Notes in Computer Science, 2009, , 258-288.	1.0	7
61	Asymptotic Resource Usage Bounds. Lecture Notes in Computer Science, 2009, , 294-310.	1.0	7
62	Inferring non-suspension conditions for logic programs with dynamic scheduling. ACM Transactions on Computational Logic, 2008, 9, 1-43.	0.7	5
63	Termination Analysis of Java Bytecode. Lecture Notes in Computer Science, 2008, , 2-18.	1.0	40
64	Automatic Inference of Upper Bounds for Recurrence Relations in Cost Analysis. Lecture Notes in Computer Science, 2008, , 221-237.	1.0	64
65	COSTA: Design and Implementation of a Cost and Termination Analyzer for Java Bytecode. Lecture Notes in Computer Science, 2008, , 113-132.	1.0	50
66	Heap space analysis for java bytecode. , 2007, , .		31
67	Termination analysis of logic programs through combination of type-based norms. ACM Transactions on Programming Languages and Systems, 2007, 29, 10.	1.7	49
68	Experiments in Cost Analysis of Java Bytecode. Electronic Notes in Theoretical Computer Science, 2007, 190, 67-83.	0.9	11
69	Detecting Determinacy in Prolog Programs. Lecture Notes in Computer Science, 2006, , 132-147.	1.0	2
70	Inferring termination conditions for logic programs using backwards analysis. Theory and Practice of Logic Programming, 2005, 5, 75-91.	1.1	8
71	Information Flow Analysis for Java Bytecode. Lecture Notes in Computer Science, 2005, , 346-362.	1.0	47
72	Reuse of Results in Termination Analysis of Typed Logic Programs. Lecture Notes in Computer Science, 2002, , 477-492.	1.0	9

#	Article	IF	Citations
73	Combining Norms to Prove Termination. Lecture Notes in Computer Science, 2002, , 126-138.	1.0	18
74	Worst-case groundness analysis using definite Boolean functions. Theory and Practice of Logic Programming, 2001, 1, 611-615.	1.1	7
75	Inferring Termination Conditions for Logic Programs Using Backwards Analysis. Lecture Notes in Computer Science, 2001, , 685-694.	1.0	16
76	Higher-Precision Groundness Analysis. Lecture Notes in Computer Science, 2001, , 135-149.	1.0	2
77	The Def-inite Approach to Dependency Analysis. Lecture Notes in Computer Science, 2001, , 417-431.	1.0	4
78	Termination Analysis of Programs with Multiphase Control-Flow. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 344, 13-21.	0.8	0