Thomas Reps

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/11310413/publications.pdf

Version: 2024-02-01

106 papers	7,770 citations	136950 32 h-index	76900 74 g-index
108 all docs	108 docs citations	108 times ranked	1520 citing authors

#	Article	IF	Citations
1	Interprocedural slicing using dependence graphs. ACM Transactions on Programming Languages and Systems, 1990, 12, 26-60.	2.1	1,134
2	Precise interprocedural dataflow analysis via graph reachability. , 1995, , .		810
3	The Cornell program synthesizer. Communications of the ACM, 1981, 24, 563-573.	4.5	590
4	Parametric shape analysis via 3-valued logic. ACM Transactions on Programming Languages and Systems, 2002, 24, 217-298.	2.1	553
5	Integrating noninterfering versions of programs. ACM Transactions on Programming Languages and Systems, 1989, 11, 345-387.	2.1	294
6	Solving shape-analysis problems in languages with destructive updating. ACM Transactions on Programming Languages and Systems, 1998, 20, 1-50.	2.1	230
7	Incremental Context-Dependent Analysis for Language-Based Editors. ACM Transactions on Programming Languages and Systems, 1983, 5, 449-477.	2.1	216
8	Precise interprocedural dataflow analysis with applications to constant propagation. Theoretical Computer Science, 1996, 167, 131-170.	0.9	216
9	Program analysis via graph reachability. Information and Software Technology, 1998, 40, 701-726.	4.4	216
10	Analyzing Memory Accesses in x86 Executables. Lecture Notes in Computer Science, 2004, , 5-23.	1.3	201
11	WYSINWYX. ACM Transactions on Programming Languages and Systems, 2010, 32, 1-84.	2.1	164
12	Analysis of recursive state machines. ACM Transactions on Programming Languages and Systems, 2005, 27, 786-818.	2.1	154
13	The synthesizer generator. ACM SIGPLAN Notices, 1984, 19, 42-48.	0.2	150
14	Weighted pushdown systems and their application to interprocedural dataflow analysis. Science of Computer Programming, 2005, 58, 206-263.	1.9	144
15	Incremental evaluation for attribute grammars with application to syntax-directed editors. , 1981, , .		138
16	The synthesizer generator. , 1984, , .		127
17	Demand interprocedural dataflow analysis. , 1995, , .		125
18	Reducing concurrent analysis under a context bound toÂsequential analysis. Formal Methods in System Design, 2009, 35, 73-97.	0.8	119

#	Article	IF	Citations
19	Speeding up slicing. , 1994, , .		117
20	Symbolic Implementation of the Best Transformer. Lecture Notes in Computer Science, 2004, , 252-266.	1.3	96
21	CodeSurfer/x86â€"A Platform for Analyzing x86 Executables. Lecture Notes in Computer Science, 2005, , 250-254.	1.3	94
22	Undecidability of context-sensitive data-dependence analysis. ACM Transactions on Programming Languages and Systems, 2000, 22, 162-186.	2.1	87
23	Interprocedural slicing using dependence graphs. ACM SIGPLAN Notices, 2004, 39, 229-243.	0.2	77
24	Recency-Abstraction for Heap-Allocated Storage. Lecture Notes in Computer Science, 2006, , 221-239.	1.3	77
25	Putting static analysis to work for verification. , 2000, , .		74
26	Interconvertibility of a class of set constraints and context-free-language reachability. Theoretical Computer Science, 2000, 248, 29-98.	0.9	73
27	Optimal-time incremental semantic analysis for syntax-directed editors. , 1982, , .		68
28	Symbolically Computing Most-Precise Abstract Operations for Shape Analysis. Lecture Notes in Computer Science, 2004, , 530-545.	1.3	65
29	Shape analysis as a generalized path problem. , 1995, , .		52
30	Numeric Domains with Summarized Dimensions. Lecture Notes in Computer Science, 2004, , 512-529.	1.3	51
31	Program specialization via program slicing. Lecture Notes in Computer Science, 1996, , 409-429.	1.3	50
32	A program integration algorithm that accommodates semantics-preserving transformations. ACM Transactions on Software Engineering and Methodology, 1992, 1, 310-354.	6.0	46
33	Speeding up slicing. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1994, 19, 11-20.	0.7	45
34	Extended Weighted Pushdown Systems. Lecture Notes in Computer Science, 2005, , 434-448.	1.3	45
35	Solving demand versions of interprocedural analysis problems. Lecture Notes in Computer Science, 1994, , 389-403.	1.3	43
36	Interprocedural Analysis of Concurrent Programs Under a Context Bound. , 2008, , 282-298.		43

#	Article	IF	Citations
37	A framework for numeric analysis of array operations. ACM SIGPLAN Notices, 2005, 40, 338-350.	0.2	42
38	Intermediate-representation recovery from low-level code. , 2006, , .		39
39	Extracting Output Formats from Executables. , 2006, , .		37
40	TSL. ACM Transactions on Programming Languages and Systems, 2013, 35, 1-59.	2.1	35
41	Model Checking of Unrestricted Hierarchical State Machines. Lecture Notes in Computer Science, 2001, , 652-666.	1.3	34
42	The why and wherefore of the Cornell Program Synthesizer. ACM SIGPLAN Notices, 1981, 16, 8-16.	0.2	33
43	Finite Differencing of Logical Formulas for Static Analysis. Lecture Notes in Computer Science, 2003, , 380-398.	1.3	33
44	Compositional recurrence analysis revisited. , 2017, , .		28
45	Bilateral Algorithms for Symbolic Abstraction. Lecture Notes in Computer Science, 2012, , 111-128.	1.3	28
46	Improving Pushdown System Model Checking. Lecture Notes in Computer Science, 2006, , 343-357.	1.3	27
47	A Relational Approach to Interprocedural Shape Analysis. Lecture Notes in Computer Science, 2004, , 246-264.	1.3	27
48	Low-Level Library Analysis and Summarization. , 2007, , 68-81.		27
49	Demand interprocedural dataflow analysis. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1995, 20, 104-115.	0.7	23
50	A relational approach to interprocedural shape analysis. ACM Transactions on Programming Languages and Systems, 2010, 32, 1-52.	2.1	23
51	Algebraic properties of program integration. Science of Computer Programming, 1991, 17, 139-215.	1.9	22
52	Synthesis of machine code from semantics. , 2015, , .		22
53	Program generalization for software reuse. , 1996, , .		21
54	The synthesizer generator. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1984, 9, 42-48.	0.7	20

#	Article	IF	Citations
55	Fast graph simplification for interleaved Dyck-reachability. , 2020, , .		20
56	Efficient comparison of program slices. Acta Informatica, 1991, 28, 713-732.	0.5	19
57	Abstract Domains of Affine Relations. ACM Transactions on Programming Languages and Systems, 2014, 36, 1-73.	2.1	18
58	On the sequential nature of interprocedural program-analysis problems. Acta Informatica, 1996, 33, 739-757.	0.5	17
59	Finding Concurrency-Related Bugs Using Random Isolation. Lecture Notes in Computer Science, 2008, , 198-213.	1.3	17
60	Program Analysis Using Weighted Pushdown Systems. Lecture Notes in Computer Science, 2007, , 23-51.	1.3	16
61	On competitive on-line algorithms for the dynamic priority-ordering problem. Information Processing Letters, 1994, 51, 155-161.	0.6	15
62	Precise interprocedural chopping. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1995, 20, 41-52.	0.7	14
63	On the sequential nature of interprocedural program-analysis problems. Acta Informatica, 1996, 33, 739-757.	0.5	14
64	Pointer analysis for programs with structures and casting. ACM SIGPLAN Notices, 1999, 34, 91-103.	0.2	14
65	Automating Abstract Interpretation. Lecture Notes in Computer Science, 2016, , 3-40.	1.3	14
66	OpenNWA: A Nested-Word Automaton Library. Lecture Notes in Computer Science, 2012, , 665-671.	1.3	14
67	Newtonian program analysis via tensor product. , 2016, , .		14
68	Verifying Information Flow Control over Unbounded Processes. Lecture Notes in Computer Science, 2009, , 773-789.	1.3	13
69	Physical type checking for C. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1999, 24, 66-75.	0.7	12
70	ConSeq. ACM SIGPLAN Notices, 2011, 46, 251-264.	0.2	12
71	Refinement of path expressions for static analysis. , 2019, 3, 1-29.		12
72	Compositional recurrence analysis revisited. ACM SIGPLAN Notices, 2017, 52, 248-262.	0.2	12

#	Article	IF	Citations
73	Semantic foundations of binding-time analysis for imperative programs. , 1995, , .		11
74	ConSeq. Computer Architecture News, 2011, 39, 251-264.	2.5	11
75	Symbolic analysis via semantic reinterpretation. International Journal on Software Tools for Technology Transfer, 2011, 13, 61-87.	1.9	11
76	A semantics for procedure local heaps and its abstractions. ACM SIGPLAN Notices, 2005, 40, 296-309.	0.2	11
77	Finding concurrency-related bugs using random isolation. International Journal on Software Tools for Technology Transfer, 2011, 13, 495-518.	1.9	10
78	Language Strength Reduction. Lecture Notes in Computer Science, 2008, , 283-298.	1.3	10
79	Abstract Error Projection. Lecture Notes in Computer Science, 2007, , 200-217.	1.3	10
80	A Decision Procedure for Detecting Atomicity Violations for Communicating Processes with Locks. Lecture Notes in Computer Science, 2009, , 125-142.	1.3	9
81	Partial evaluation of machine code. , 2015, , .		9
82	Coping with type casts in C. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1999, 24, 180-198.	0.7	8
83	Specialization Slicing. ACM Transactions on Programming Languages and Systems, 2014, 36, 1-67.	2.1	8
84	Interconvertbility of set constraints and context-free language reachability. ACM SIGPLAN Notices, 1997, 32, 74-89.	0.2	8
85	Safety checking of machine code. ACM SIGPLAN Notices, 2000, 35, 70-82.	0.2	7
86	On the complexity of bidirected interleaved Dyck-reachability. , 2021, 5, 1-28.		7
87	Abstract Domains of Affine Relations. Lecture Notes in Computer Science, 2011, , 198-215.	1.3	7
88	PMAF: an algebraic framework for static analysis of probabilistic programs. ACM SIGPLAN Notices, 2018, 53, 513-528.	0.2	6
89	Automatic Assume/Guarantee Reasoning for Heap-Manipulating Programs. Electronic Notes in Theoretical Computer Science, 2005, 131, 125-138.	0.9	5
90	Algebraic Program Analysis. Lecture Notes in Computer Science, 2021, , 46-83.	1.3	5

#	Article	IF	Citations
91	Solving Multiple Dataflow Queries Using WPDSs. Lecture Notes in Computer Science, 2008, , 93-109.	1.3	5
92	Static Analysis of Software Executables. , 2009, , .		4
93	View-Augmented Abstractions. Electronic Notes in Theoretical Computer Science, 2010, 267, 43-57.	0.9	4
94	Newtonian Program Analysis via Tensor Product. ACM Transactions on Programming Languages and Systems, 2017, 39, 1-72.	2.1	4
95	Sublinear-space evaluation algorithms for attribute grammars. ACM Transactions on Programming Languages and Systems, 1987, 9, 408-440.	2.1	3
96	A decision procedure for detecting atomicity violations for communicating processes with locks. International Journal on Software Tools for Technology Transfer, 2011, 13, 37-60.	1.9	3
97	Sound Bit-Precise Numerical Domains. Lecture Notes in Computer Science, 2017, , 500-520.	1.3	3
98	Fast Graph Simplification for Interleaved-Dyck Reachability. ACM Transactions on Programming Languages and Systems, 2022, 44, 1-28.	2.1	3
99	A Next-Generation Platform for Analyzing Executables. , 2007, , 43-61.		2
100	Analysis Techniques for Information Security. Synthesis Lectures on Information Security Privacy and Trust, 2010, 2, 1-164.	0.3	2
101	A new abstraction framework for affine transformers. Formal Methods in System Design, 2019, 54, 110-143.	0.8	2
102	Newtonian program analysis via tensor product. ACM SIGPLAN Notices, 2016, 51, 663-677.	0.2	2
103	The why and wherefore of the Cornell Program Synthesizer. ACM SIGOA Newsletter, 1981, 2, 8-16.	0.1	1
104	Interprocedural Context-Unbounded Program Analysis Using Observation Sequences. ACM Transactions on Programming Languages and Systems, 2020, 42, 1-34.	2.1	0
105	Partial evaluation of machine code. ACM SIGPLAN Notices, 2015, 50, 860-879.	0.2	0
106	Incremental evaluation for attribute grammars with unrestricted movement between tree modifications. Acta Informatica, 1988, 25, 155-178.	0.5	0