

Andrew W Appel

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/482741/publications.pdf>

Version: 2024-02-01

109
papers

4,484
citations

257429

24
h-index

182417

51
g-index

116
all docs

116
docs citations

116
times ranked

1167
citing authors

#	ARTICLE	IF	CITATIONS
1	An Efficient Program for Many-Body Simulation. SIAM Journal on Scientific and Statistical Computing, 1985, 6, 85-103.	1.5	452
2	Simple generational garbage collection and fast allocation. Software - Practice and Experience, 1989, 19, 171-183.	3.6	234
3	An indexed model of recursive types for foundational proof-carrying code. ACM Transactions on Programming Languages and Systems, 2001, 23, 657-683.	2.1	222
4	Iterated register coalescing. ACM Transactions on Programming Languages and Systems, 1996, 18, 300-324.	2.1	197
5	Virtual memory primitives for user programs. , 1991, , .		160
6	SSA is functional programming. ACM SIGPLAN Notices, 1998, 33, 17-20.	0.2	122
7	Standard ML of New Jersey. Lecture Notes in Computer Science, 1991, , 1-13.	1.3	111
8	Garbage collection can be faster than stack allocation. Information Processing Letters, 1987, 25, 275-279.	0.6	109
9	SAFKASI. ACM Transactions on Software Engineering and Methodology, 2000, 9, 341-378.	6.0	106
10	Verification of a Cryptographic Primitive. ACM Transactions on Programming Languages and Systems, 2015, 37, 1-31.	2.1	97
11	A semantic model of types and machine instructions for proof-carrying code. , 2000, , .		93
12	A very modal model of a modern, major, general type system. , 2007, , .		88
13	A Fresh Look at Separation Algebras and Share Accounting. Lecture Notes in Computer Science, 2009, , 161-177.	1.3	86
14	Verified Software Toolchain. Lecture Notes in Computer Science, 2011, , 1-17.	1.3	80
15	Space-efficient closure representations. , 1994, , .		64
16	A type-based compiler for standard ML. , 1995, , .		61
17	VST-Floyd: A Separation Logic Tool to Verify Correctness of C Programs. Journal of Automated Reasoning, 2018, 61, 367-422.	1.4	56
18	Runtime tags aren't necessary. Higher-Order and Symbolic Computation, 1989, 2, 153-162.	0.6	55

#	ARTICLE	IF	CITATIONS
19	Oracle Semantics for Concurrent Separation Logic. , 2008, , 353-367.		55
20	Real-time concurrent collection on stock multiprocessors. ACM SIGPLAN Notices, 2004, 39, 205-216.	0.2	54
21	Compositional CompCert. , 2015, , .		51
22	The world's fastest Scrabble program. Communications of the ACM, 1988, 31, 572-578.	4.5	47
23	A runtime system. Higher-Order and Symbolic Computation, 1990, 3, 343-380.	0.6	45
24	Smartest recompilation. , 1993, , .		41
25	Debugging standard ML without reverse engineering. , 1990, , .		39
26	Shrinking lambda expressions in linear time. Journal of Functional Programming, 1997, 7, 515-540.	0.8	39
27	Hierarchical modularity. ACM Transactions on Programming Languages and Systems, 1999, 21, 813-847.	2.1	38
28	Empirical and analytic study of stack versus heap cost for languages with closures. Journal of Functional Programming, 1996, 6, 47-74.	0.8	36
29	Separation Logic for Small-Step cminor. Lecture Notes in Computer Science, 2007, , 5-21.	1.3	36
30	Type-preserving garbage collectors. , 2001, , .		35
31	A Compositional Logic for Control Flow. Lecture Notes in Computer Science, 2005, , 80-94.	1.3	34
32	A Debugger for Standard ML. Journal of Functional Programming, 1995, 5, 155-200.	0.8	33
33	Verified Compilation for Shared-Memory C. Lecture Notes in Computer Science, 2014, , 107-127.	1.3	33
34	Foundational proof checkers with small witnesses. , 2003, , .		31
35	Verified Correctness and Security of mbedTLS HMAC-DRBG. , 2017, , .		30
36	An advisor for flexible working sets. , 1990, , .		29

#	ARTICLE	IF	CITATIONS
37	Semantic foundations for typed assembly languages. ACM Transactions on Programming Languages and Systems, 2010, 32, 1-67.	2.1	29
38	A provably sound TAL for back-end optimization. , 2003, , .		23
39	Efficient and safe-for-space closure conversion. ACM Transactions on Programming Languages and Systems, 2000, 22, 129-161.	2.1	22
40	A Trustworthy Proof Checker. Journal of Automated Reasoning, 2003, 31, 231-260.	1.4	21
41	A very modal model of a modern, major, general type system. ACM SIGPLAN Notices, 2007, 42, 109-122.	0.2	21
42	Position paper: the science of deep specification. Philosophical Transactions Series A, Mathematical, Physical, and Engineering Sciences, 2017, 375, 20160331.	3.4	21
43	Virtual memory primitives for user programs. ACM SIGPLAN Notices, 1991, 26, 96-107.	0.2	20
44	Separate compilation for Standard ML. , 1994, , .		20
45	A theory of indirection via approximation. , 2010, , .		20
46	A Certificate Infrastructure for Machine-Checked Proofs of Conditional Information Flow. Lecture Notes in Computer Science, 2012, , 369-389.	1.3	19
47	Callee-save registers in continuation-passing style. Higher-Order and Symbolic Computation, 1992, 5, 191-221.	0.6	18
48	Cache performance of fast-allocating programs. , 1995, , .		18
49	Iterated register coalescing. , 1996, , .		17
50	Mechanisms for secure modular programming in Java. Software - Practice and Experience, 2003, 33, 461-480.	3.6	17
51	A verified messaging system. , 2017, 1, 1-28.		17
52	A critique of Standard ML. Journal of Functional Programming, 1993, 3, 391-429.	0.8	15
53	Dependent types ensure partial correctness of theorem provers. Journal of Functional Programming, 2004, 14, 3-19.	0.8	15
54	Generalizations of the sethi-ullman algorithm for register allocation. Software - Practice and Experience, 1987, 17, 417-421.	3.6	14

#	ARTICLE	IF	CITATIONS
55	Optimal spilling for CISC machines with few registers. ACM SIGPLAN Notices, 2001, 36, 243-253.	0.2	14
56	Security Seals on Voting Machines. ACM Transactions on Information and System Security, 2011, 14, 1-29.	4.5	14
57	Compositional CompCert. ACM SIGPLAN Notices, 2015, 50, 275-287.	0.2	14
58	Semantics-directed code generation. , 1985, , .		13
59	Formal Verification of Coalescing Graph-Coloring Register Allocation. Lecture Notes in Computer Science, 2010, , 145-164.	1.3	12
60	Closure conversion is safe for space. , 2019, 3, 1-29.		12
61	Portable Software Fault Isolation. , 2014, , .		11
62	Concurrent Separation Logic for Pipelined Parallelization. Lecture Notes in Computer Science, 2010, , 151-166.	1.3	11
63	VeriSmall: Verified Smallfoot Shape Analysis. Lecture Notes in Computer Science, 2011, , 231-246.	1.3	11
64	Lambda-splitting. , 1997, , .		10
65	Multimodal Separation Logic for Reasoning About Operational Semantics. Electronic Notes in Theoretical Computer Science, 2008, 218, 5-20.	0.9	10
66	Bringing Order to the Separation Logic Jungle. Lecture Notes in Computer Science, 2017, , 190-211.	1.3	10
67	Construction of a Semantic Model for a Typed Assembly Language. Lecture Notes in Computer Science, 2004, , 30-43.	1.3	10
68	Loop headers in λ -calculus or CPS. Higher-Order and Symbolic Computation, 1994, 7, 337-343.	0.6	9
69	Vectorized garbage collection. Journal of Supercomputing, 1989, 3, 151-160.	3.6	8
70	Axiomatic bootstrapping. ACM Transactions on Programming Languages and Systems, 1994, 16, 1699-1718.	2.1	8
71	Compositional optimizations for CertiCoq. , 2021, 5, 1-30.		8
72	A type-based compiler for standard ML. ACM SIGPLAN Notices, 1995, 30, 116-129.	0.2	7

#	ARTICLE	IF	CITATIONS
73	Allocation without locking. Software - Practice and Experience, 1989, 19, 703-705.	3.6	6
74	Unrolling lists. ACM SIGPLAN Lisp Pointers, 1994, VII, 185-195.	0.1	6
75	Space-efficient closure representations. ACM SIGPLAN Lisp Pointers, 1994, VII, 150-161.	0.1	6
76	Hoare logic. , 0, , 10-15.		6
77	Shrink fast correctly!. , 2017, , .		6
78	Abstraction and Subsumption in Modular Verification of C Programs. Lecture Notes in Computer Science, 2019, , 573-590.	1.3	6
79	Creating and preserving locality of java applications at allocation and garbage collection times. ACM SIGPLAN Notices, 2002, 37, 13-25.	0.2	6
80	Debuggable concurrency extensions for standard ML. ACM SIGPLAN Notices, 1991, 26, 120-131.	0.2	5
81	A provably sound TAL for back-end optimization. ACM SIGPLAN Notices, 2003, 38, 208-219.	0.2	5
82	Polymorphic lemmas and definitions in λ Prolog and Twelf. Theory and Practice of Logic Programming, 2004, 4, 1-39.	1.5	5
83	Verified heap theorem prover by paramodulation. , 2012, , .		5
84	A List-Machine Benchmark for Mechanized Metatheory. Journal of Automated Reasoning, 2012, 49, 453-491.	1.4	5
85	Modular Verification for Computer Security. , 2016, , .		5
86	Connecting Higher-Order Separation Logic to a First-Order Outside World. Lecture Notes in Computer Science, 2020, , 428-455.	1.3	5
87	Verified heap theorem prover by paramodulation. ACM SIGPLAN Notices, 2012, 47, 3-14.	0.2	5
88	Verified sequential Malloc/Free. , 2020, , .		5
89	A List-machine Benchmark for Mechanized Metatheory. Electronic Notes in Theoretical Computer Science, 2007, 174, 95-108.	0.9	4
90	Abstraction and subsumption in modular verification of C programs. Formal Methods in System Design, 2021, 58, 322-345.	0.8	4

#	ARTICLE	IF	CITATIONS
91		0.2	3
92	A theory of indirection via approximation. ACM SIGPLAN Notices, 2010, 45, 171-184.	0.2	3
93	Verification of a cryptographic primitive: SHA-256 (abstract). , 2015, , .		3
94	Mostly Sound Type System Improves a Foundational Program Verifier. Lecture Notes in Computer Science, 2013, , 17-32.	1.3	3
95	Coq's vibrant ecosystem for verification engineering (invited talk). , 2022, , .		3
96	Viewpoint: Technological access control interferes with noninfringing scholarship. Communications of the ACM, 2000, 43, 21-23.	4.5	2
97	Local actions for a curry-style operational semantics. , 2011, , .		2
98	Deriving efficient program transformations from rewrite rules. , 2021, 5, 1-29.		2
99	Is POPL mathematics or science?. ACM SIGPLAN Notices, 1992, 27, 87-89.	0.2	2
100	Virtual memory primitives for user programs. Computer Architecture News, 1991, 19, 96-107.	2.5	1
101	Special Issue on ML. Journal of Functional Programming, 1993, 3, 389-389.	0.8	1
102	Type-preserving garbage collectors. ACM SIGPLAN Notices, 2001, 36, 166-178.	0.2	1
103	Efficient Substitution in Hoare Logic Expressions. Electronic Notes in Theoretical Computer Science, 2001, 41, 35-49.	0.9	1
104	Separate compilation for Standard ML. ACM SIGPLAN Notices, 1994, 29, 13-23.	0.2	1
105	An advisor for flexible working sets. Performance Evaluation Review, 1990, 18, 153-162.	0.6	0
106	Virtual memory primitives for user programs. Operating Systems Review (ACM), 1991, 25, 96-107.	1.9	0
107	Special Issue on ML. Journal of Functional Programming, 1992, 2, i-i.	0.8	0
108	Lambda-splitting. ACM SIGPLAN Notices, 1997, 32, 112-124.	0.2	0

#	ARTICLE	IF	CITATIONS
109	A Logical Mix of Approximation and Separation. Lecture Notes in Computer Science, 2010, , 439-454.	1.3	0